



Edition 2.5

MERANT® TRACKER™
ADMINISTRATOR'S GUIDE

FROM THE MAKERS OF PVCS®

Copyright © 1992–2004 Merant. All rights reserved.

Merant® and PVCS® are registered trademarks of Merant, and Merant Dimensions, Merant Mover, Merant Professional, Merant Meritage, Merant Tracker, Merant Version Manager, Merant Collage, Merant Build, Merant Modello, Merant Metrics, Merant Pulse, Merant TrackerLink, and PVCS Change Manager for Oracle are trademarks of Merant. All other trademarks are the property of their respective owners.

Portions of this software are copyrighted by DataDirect Technologies, 1991–2004.

No part of this publication may be copied, photocopied, reproduced, transmitted, transcribed, or reduced to any electronic medium or machine-readable form without prior written consent of Merant.

Licensee may copy the software product user documentation contained on a CD-ROM, but only to the extent necessary to support the users with authorized access to the software in accordance with the license agreement applicable to the software. Any reproduction of such software product user documentation, regardless of whether the documentation is reproduced in whole or in part, must be accompanied by this copyright statement in its entirety, without modification.

U.S. GOVERNMENT RESTRICTED RIGHTS. It is acknowledged that the Software and the Documentation were developed at private expense, that no part is in the public domain, and that the Software and Documentation are Commercial Computer Software provided with RESTRICTED RIGHTS under Federal Acquisition Regulations and agency supplements to them. Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in the license agreement accompanying such Software and Documentation and as provided in, among other sources, DFARS 252.227-7015 and 227.7202, or subparagraphs (c)(1) and (2) of the Commercial Computer Software Restricted Rights at FAR 52.227-19, as applicable. Contractor is Merant, 3445 NW 211th Terrace, Hillsboro, Oregon 97124. Rights are reserved under copyright laws of the United States with respect to unpublished portions of the Software.

Merant
3445 NW 211th Terrace
Hillsboro, Oregon 97124 U.S.A.

MA-TRKADMIN-002

Table of Contents

Welcome to Tracker	13
Typographical Conventions.	14
Ordering Hard-Copy Manuals.	15
Contacting Technical Support.	15
1 Introduction to Tracker Administrator.	17
Introducing Tracker Administrator.	18
What is the Tracker Web Client?.	18
Components of Tracker	22
Database Schema.	23
Exploring the Tracker Administrator Interface	25
Toolbar Buttons	26
Manage Menu	26
Changing View Options	26
Managing Task Windows	27
Closing Task Windows	27
Glossary	28
2 Implementing Tracker	37
How Tracker Fits Into Your DBMS Environment	38
People Involved In Managing a Tracker Project	39
Four Types of Administrators	39
Server Administrator Tasks	41
Project Administrator Tasks.	42
Planning a Successful Tracker Implementation	44
Size of Your Organization.	44
Tracker's Impact on the QA Environment	45

	Tracker Implementation Planning Checklists	47
3	Logging In to Tracker Administrator	55
	Overview	56
	Logging In to Tracker Administrator	56
	Practicing with the Learning Projects	58
	Switching to Another Project	59
	Logging Out of Tracker Administrator	59
4	Preparing the Server	61
	Overview	62
	Elements of Server Preparation	63
	About Tracker Databases	64
	About Devices, Tablespace, and Datasources	64
	Creating Devices and Tablespaces	65
	About the Tracker-DBMS Login Relationship	66
	For Tracker Users	66
	For Tracker Administrators	67
	Information You Need Before You Begin	68
	Working with Server Definitions	72
	The Tracker Server Wizard	72
	Converting Old Server Definitions	72
	Creating New Server Definitions	73
	Editing Server Definitions	76
	Deleting Server Definitions	76
	Making Server Definitions Available to All Users	77
	Preparing the Server	79
5	Creating and Managing Projects	85
	Overview	86

Elements of Project Creation	87
About Projects	87
Devices and Tablespaces	88
Creating a Project	89
Upgrading Your Projects	92
When You Upgrade	92
If You Don't Upgrade	93
Before You Upgrade	93
How the Upgrade Affects the State Field	96
How to Upgrade Your Projects	97
Reattaching an Existing Database	98
Creating a New Project Based on an Existing One	100
Allowing Special System Permissions for Project Administrators	101
Renaming an Existing Project	102
Deleting a Project	103
6 Creating and Managing Users	105
Overview	106
About Users	106
Adding Users	107
Adding Users to the DBMS	108
DBA User Creation Tasks	109
Adding Users in Tracker	110
Setting Up Notify Options for a User	112
Setting Up Mail Options for a User	114
Setting Up Project Administrators	116
Modifying Users	116
Deleting Users	117
Recovering Deleted Users	118

Copying Users	119
Enabling Single Sign-On for Users.	120
7 Setting up LDAP for Login Authentication	121
Overview	122
Using LDAP in Tracker	122
Configuring LDAP in Tracker	122
Configuring Authentication	123
Configuring LDAP Servers	126
Configuring LDAP Users.	128
Setting Up Redundant and Distributed Servers	129
8 Managing User Groups and Permissions.	131
Overview	132
About User Groups and Permissions	132
Standard User Groups	133
Selecting Appropriate User Group Permissions	134
Available Permissions	135
Modifying User Group Permissions	135
Selecting a User Group to Modify	136
Setting General Permissions	137
Setting Record Permissions	139
Setting Field Permissions	143
Setting Query Permissions	145
Setting Report Permissions	147
Creating User Groups.	149
Creating Token User Groups	150
Deleting User Groups.	151
Copying User Groups	152
9 Preparing Data for External Reporting Tools	155
Overview	156

What the Assistant Does for You	156
Required Skills and Knowledge	157
ODBC Knowledge	157
Database Report Writing Skills	158
Familiarity with Your Project Database	158
Choosing a Database Querying and Reporting Tool	159
Determining the Kinds of Reports You Want to Create ..	159
Using the Assistant Under MS SQL Server	160
Database Permissions Required	160
Creating Views and Tables Under MS SQL	161
Using the Assistant Under Sybase	166
Database Permissions Required	166
Determine Whether You Should Create Views or Tables ..	166
Creating Views and Tables Under Sybase	167
Using the Assistant Under Oracle	173
Database Permissions Required	173
Determining Whether You Should Create Views or Tables.	173
Creating Views and Tables Under Oracle	174
Creating Project Lists in the Report Assistant	179
Updating Views or Tables	180
Descriptions of the Views and Tables	180
Change Request View or Table	180
Change History View or Table	182
10 Creating Issue Types and Relationships	185
Overview	186
About Issue Types	186
Defining Multiple Issue Types	187
Adding and Modifying Issue Types	188
Setting Permissions for Issue Types	191

Using Issue Types with Dependent Fields and Transition Relationships.	191
Defining Issue Type Transition Lifecycles	192
Sharing Issue Types with System Issue Types.	193
Deleting Issue Types	195
About Issue Relationships	197
Default Relationship Types	198
Understanding Relationship Attributes	199
Defining Relationship Types	200
Creating Cross-Project Relationship Types	201
Deleting Relationship Types.	205
Setting Relationship Permissions.	205
11 Customizing Forms, Fields, and Terms	209
Overview	210
About Fields	210
Tracker Standard Fields	211
Custom Fields	211
Data Types Available	212
Custom Fields Indexing	212
Transition Fields	213
Limitations on Using Fields	213
Adding Fields	214
Defining Choice Fields	215
Defining Transition Fields	219
Defining Date Fields.	220
Defining Time Fields.	221
Defining Number Fields	223
Defining String Fields.	225
Defining User Fields	226
About Dependent Field Relationships	228
Creating Field Dependencies.	229
Modifying Fields.	231

Modifying Field Dependences	232
Deleting Fields	232
Recovering Deleted Fields	233
Copying Fields	234
Preparing Fields for Cloning	235
About Forms	236
Creating Forms	237
Modifying Forms	240
Managing Note Titles	242
Adding Note Titles	243
Modifying Note Titles	243
Deleting Note Titles	244
Controlling Record Numbering	244
Customizing Terminology	245
Customizing Terms	246
Resetting Terms to Their Default Value	247
Custom Terms and the Tracker Web Client	248
12 Integrating Tracker with Source Control	249
Overview	250
Benefits of Integrating Tracker and Source Control	251
About Version Manager Configuration Files	251
Embedding a Master Configuration File	252
About Change Text	253
How Change Text is Defined	253
Customizing Source Control Setup	254
13 Setting Up Notify	257
About Notify	258
Notify Concepts	258

- Notify Options 260
- Using Notify 262
 - Starting Notify 262
 - Configuring Global Properties. 264
 - Configuring Server Properties 266
 - Configuring Your Mail System. 266
 - Setting Up Mail Message Options..... 267
- 14 Tracker Project Management and Recovery 271**
 - Overview 272
 - Performing Data Backups and Recovery..... 272
 - Backup Recommendations. 272
 - Backing Up Tracker Data 273
 - Recovering Data from Backups 273
 - Re-Indexing Database Tables..... 274
 - Preserving Database Integrity 275
 - Critical Verify and Repair (Level 1 Corruption)..... 276
 - Verify and Repair (Level 2 Corruption) 278
 - Permitting Repairs by Project Administrators..... 281
 - Moving an Existing Database 283
 - Rolling Over an Existing Project Database 284
 - Controlling Record Locking 284
 - Turning Record Locking On or Off 285
 - Controlling Logins 286
 - Turning Off Logins 287
 - Turning On Logins 288
 - Forcing Users Off Tracker..... 288
 - Removing User Processes 289
 - Improving Query Performance 290

Appendix A: Disk Space Allocation	293
Appendix B: Default Directory Structure	295
Appendix C: Evaluating Tracker Using MSDE	297
Appendix D: Configuring the Web Client Interface . . .	299
The SAMPLE.HTM File	299
The WTSETUP.EXE File	300
Index	301

Welcome to Tracker

Thank you for choosing Merant™ Tracker™ Administrator, the administration tool for Tracker.

Purpose of this manual

This manual contains conceptual and “how to” information on using Tracker Administrator.

For more information

Refer to the *Merant Tracker User's Guide* for a description of the Tracker documentation set, a summary of the ways to work with Tracker, and instructions for accessing the online help.

Edition status

This is Edition 2.5 of the *Tracker Administrator's Guide*. The information in this edition applies to *Release 8.0 of Tracker* or later. This edition supersedes earlier editions of this manual.

Typographical Conventions

The following typographical conventions are used in the online manuals and online help. These typographical conventions are used to assist you when you use the documentation; they are not meant to contradict or change any standard use of typographical conventions in the various product components or the host operating system.

Convention	Explanation
<i>italics</i>	Introduces new terms that you may not be familiar with and occasionally indicates emphasis.
bold	Emphasizes important information and field names.
UPPERCASE	Indicates keys or key combinations that you can use. For example, press the ENTER key.
<code>monospace</code>	Indicates syntax examples, values that you specify, or results that you receive.
<i>monospaced italics</i>	Indicates names that are placeholders for values you specify; for example, <i>filename</i> .
<code>monospace bold</code>	Indicates the results of an executed command.
vertical rule	Separates menus and their associated commands. For example, select File Copy means to select Copy from the File menu. Also, indicates mutually exclusive choices in a command syntax line.
brackets []	Indicates optional items. For example, in the following statement: <code>SELECT [DISTINCT] ,</code> DISTINCT is an optional keyword.
...	Indicates command arguments that can have more than one value.

Ordering Hard-Copy Manuals

As part of your Tracker license agreement, you may print and distribute as many copies of the Tracker manuals as needed.

If you do not want to print each of these online manuals, you can order hard-copy versions from Merant. To order, please contact your sales representative for assistance.

Contacting Technical Support

Merant provides technical support for all registered users of this product, including limited installation support for the first 30 days. If you need support after that time, contact Merant using one of the methods listed in the *Installation Guide*, the *Tracker User's Guide* or the online help.

Technical support is available 24 hours a day, 7 days a week, with language-specific support available during local business hours. For all other hours, technical support is provided in English.

Support via the web, e-mail, and telephone

SupportNet Customers can report problems and ask questions on the SupportNet web page: <http://support.merant.com/>

To submit an issue, click the **Report a Problem** link and follow the instructions. You can also submit issues via e-mail or phone. Refer to the *Merant Tracker Installation Guide*, *Merant Tracker User's Guide*, or online help for a list of contact numbers, including numbers to call for local language support.

The SupportNet Web site contains up-to-date technical support information. Our SupportNet Community shares information via the Web, automatic e-mail notification, newsgroups, and regional user groups.

SupportNet Online is our global service network that provides access to valuable tools and information for an online community for users. SupportNet Online also includes a KnowledgeBase, which contains how-to information and allows you to search on keywords for technical bulletins. You can also download fix releases for your Merant products.

1 Introduction to Tracker Administrator

In this Chapter

For information about . . .	See page . . .
Introducing Tracker Administrator	18
Database Schema	23
Exploring the Tracker Administrator Interface	25
Glossary	28

Introducing Tracker Administrator

Welcome to Merant Tracker Administrator, the administrator's interface to the market-leading system change request tracking tool running on Windows and the Web. Tracker is typically used by application developers, quality assurance testers, and technical support personnel to efficiently manage system change requests for software projects. Tracker provides secure, controlled access to project databases that organize and automate the development and test environment.

Tracker Administrator creates and maintains the project environment for Tracker users. This includes creating projects, forms, fields, users, user groups, and assigning users to user groups. In addition, the Tracker Server Administrator can run certain database and server maintenance programs. For more information about the different kinds of administrator roles in Tracker, see [“People Involved In Managing a Tracker Project” on page 39](#).

The Tracker Server Administrator is also responsible for making additional Tracker components, such as the Tracker web client, available to Tracker users.

What is the Tracker Web Client?

The Tracker web client is an optional Tracker component that allows users to access most Tracker functionality via a web browser interface. You can use this new component to expand access to Tracker for users on the Internet and intranet.

- Tracker web client users do not need to install any part of the Tracker application on the local workstation. The only requirement for use of the web client interface is that a supported web browser is installed on the user's machine.

Supported web browsers are:

- Netscape Communicator
- Microsoft Internet Explorer

Functionality supported by the web client component includes:

- Viewing the In Tray
- Running Query by ID and named queries
- Creating queries
- Submitting change requests
- Updating change requests
- Adding notes
- Attaching files
- Notifying users of issues
- Cloning issues
- Printing issues and query results
- Creating relationships between issues
- Anonymous login via a shared ID

All of the project setup and most of the project configuration for the web client is done in the Tracker Administrator interface in Windows. Project configuration in the web client interface is limited to the following:

- Enabling the project to be viewed from the web

A project can be completely configured in the web client, but not enabled to be viewed from the web. This has the effect of “turning the project off” for web client users.

- Providing access to the following features:

- Submit
- Update
- Record detail
- Queries

- In Tray

Users with access to the web client interface cannot place anything in their own In Trays. Issues appear in the web client In Trays via manual notification (by desktop client or web client users) or via notification options set in the desktop client or in Tracker Administrator. Users can use the same ID and password to log in to both interfaces, so those with access to the desktop client interface can place change requests in their own In Trays manually or by using Notify options.

- Choosing custom forms created via the Tracker Administrator interface

The Tracker Server Administrator can create additional custom forms. This allows you to have separate forms for web client and desktop client interface users. For instance, you may choose to have a web client submit form. As in the case of standard forms, the Tracker Server Administrator is responsible for defining the available fields. The Web Client Administrator may then select from that set of fields, but not add additional fields in the web client.

- Setting up access for anonymous users

Once the user ID and password for anonymous login have been configured in the Tracker Administrator (see below), you need to identify that user ID for each project viewable via the web client for which anonymous login will be allowed.

The following configuration tasks must be done in the desktop client interface:

- Creating projects

- Adding new users and user groups

This includes defining an anonymous user ID if you are going to allow anonymous submissions. Anonymous users are a special type of user that exists only in the web client, though whatever user ID you choose for them to use must be set up in

the Tracker Administrator interface. This is a single user ID shared by all users who log in anonymously. It is limited to web client submit functionality only.

It's important to configure this user ID as a member of only one user group and to ensure that the user group has only submit privileges. Assigning more privileges could result in anonymous users accessing more than just Tracker's submit functionality. They would then be able to manipulate your project database in ways you did not intend. Assigning anonymous users more than submit capability is also a violation of your Tracker license agreement.

■ Configuring queries

Web client users can create custom queries (as desktop client interface users can) as well as use query by ID, run public queries, and run their own saved private queries (if they also have access to the desktop client interface).

For more information on setting up and using this component, see the *Merant Tracker Installation Guide* and the Tracker web client help system. You must have already installed Tracker and the web client component on a web server system to access the online help.

To access the Tracker web client help system:

- 1 Start the Tracker web client.
- 2 Click the Help tab. The Help Topics icon appears.
- 3 Click the Help Topics icon. The Tracker web client online help window appears.
- 4 In the page that opens, click Tracker Web Configuration.
- 5 In the Tracker web client page that opens, click the Help button in the side menu.
- 6 In the help window, click Contents; then choose "Setting Up a New Tracker Web Client Project."

Components of Tracker

Tracker is organized into the following components:

<i>Table 1-1. Component List</i>		
This component ...	Is used to ...	And is used by ...
Tracker	Submit system change requests. Create and run queries and reports.	Everyone
Tracker Administrator	Set up the Tracker database, create and manage projects, and manage Tracker users. Project administrators can use this program to perform administrative tasks on specific projects.	Database and/or Server Administrators
Tracker Web Client	Provide an optional web browser interface to Tracker projects.	Remote and mobile users on the Internet, local users on the intranet

Table 1-1. Component List *(cont.)*

This component ...	Is used to ...	And is used by ...
TrackerLink	Provides access to Tracker features from within your integrated development environment.	Programmers whose main focus is generating code inside an Independent Development Environment (IDE)
Notify	Send automatic notification of Tracker system change requests via e-mail.	Server Administrator

Database Schema

The *Tracker Database Schema Specification* is found in the online help. The schema is a description of all the Tracker tables and their uses. This document is provided to help database administrators to access information in the database through the use of third-party reporting and querying tools.

NOTE Merant does not recommend changing the values of records in Tracker databases through the use of third-party tools.



Sample Scenario

Recently Brad Churchill's boss called him into a special one-on-one to deliver the news that the company was buying Tracker to streamline the QA function and that Brad would be the new Tracker Server Administrator. In this role, Brad found that he would be interacting frequently with Denise Swinburn, the SQL Server Database Administrator (DBA). Brad was pleased with this: database administration would be one less thing he'd have to do. His boss handed him the software manual and installation disks.

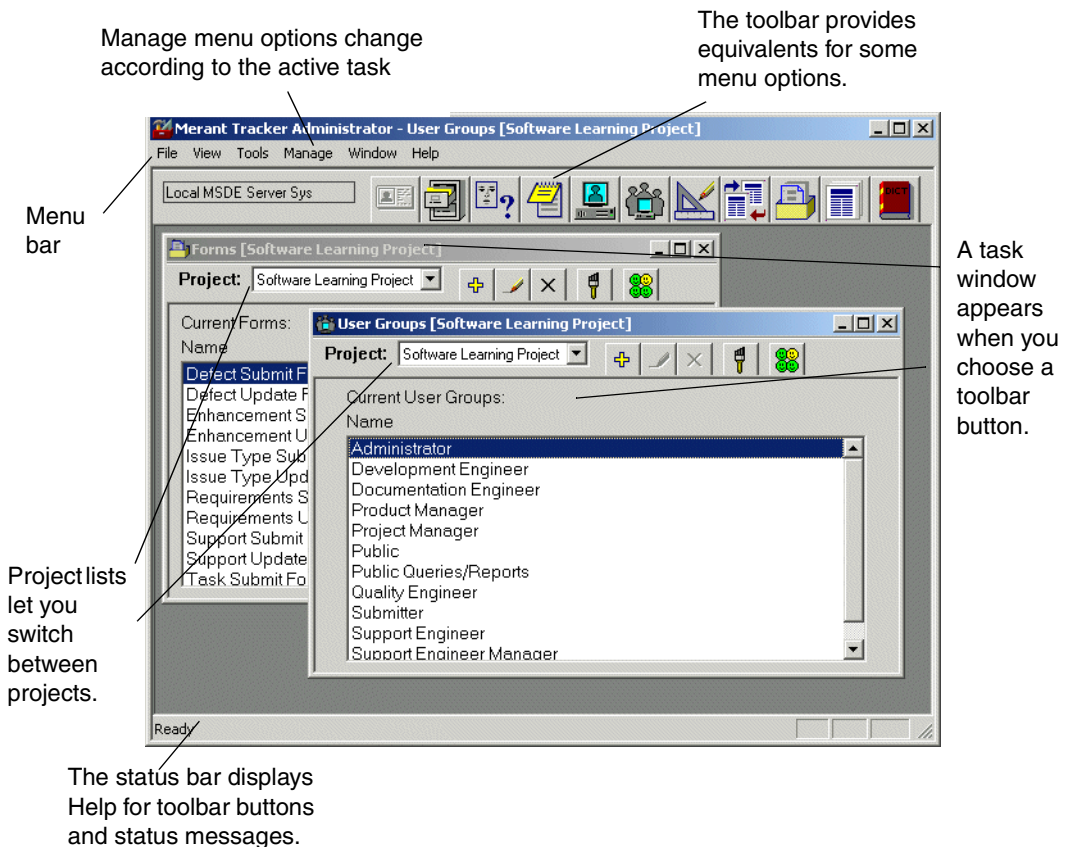
"This is an out-of-the-box solution. Go make it happen and get back to me when things are running perfectly!" his boss ordered.

Brad returned to his office and started reading "[Introduction to Tracker Administrator](#)," to get an understanding of what Tracker Administrator is. (See page [43](#) for a continuation of this scenario.)

Exploring the Tracker Administrator Interface

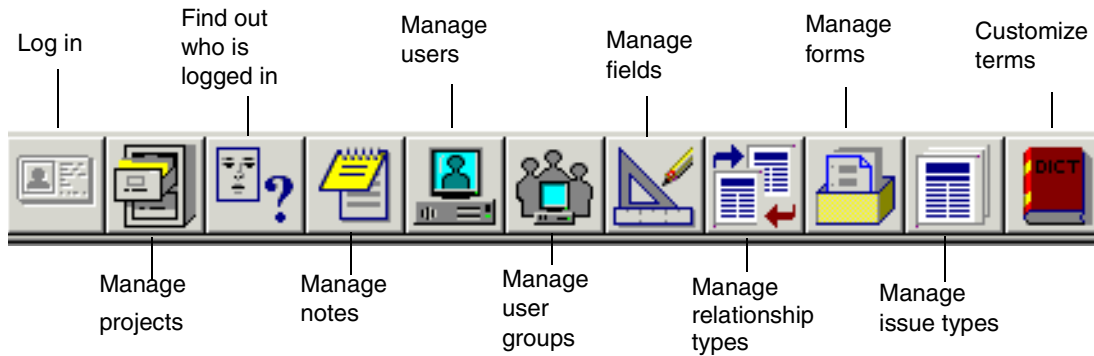
Most of the work you do in Tracker Administrator will be done in the main window. This window allows you to switch from mode to mode using the toolbar (see [“Toolbar Buttons” on page 26](#)) and accomplish administrative tasks using the menus (see [“Manage Menu” on page 26](#)).

The graphic below shows the components of the Tracker Administrator main window.



Toolbar Buttons

The toolbar provides quick access to Tracker Administrator functions:



Manage Menu

The Manage menu in Tracker Administrator changes depending on the activity task you are working in. For more information on each menu option, search for the menu option in Help | Search.

Changing View Options

To control whether the toolbar is displayed, select View | Tool Bar.
To control whether the status bar and status bar help are displayed, select View | Status Bar.

Managing Task Windows

Tracker Administrator provides several tools for arranging task windows in the main window:

- To arrange the task windows in a side-by-side pattern, select Window | Tile.
- To arrange task windows in an overlapping cascade pattern, select Window | Cascade.
- To arrange the icons inside a minimized task window, select Window | Arrange Icon.

Closing Task Windows

To close a task window, do one of the following:

- Double-click the system menu symbol at the top left corner of the window.
- Activate the task window; then select File | Close from the main menu bar.

Glossary

Admin user ID	A special Tracker user ID in every project database. The <i>Admin</i> user ID cannot be deleted. The Administrator user group always has all permissions.
Admin user	Tracker users who are members of the Administrator group and can administrate and manage projects.
Conditional Notify	Conditional Notify is an enhancement to Notify that makes it possible for you to be notified when a particular field changes from one choice to another, such as when a Priority changes from High to Urgent.
custom field	A field in an issue that you define specifically for the type of data you track in your working environment. There are six types of custom fields: choice, date, number, string, time, and user.
database device	The database device where you will store data for the project when using Sybase.
datasources	The datasource is where you will store data for the project when using SQL Server.
DBA	Database administrator. The person who manages a DBMS, and has full administrative privileges to all databases in the system.
DBMS	Database Management System. This is a standard industry term for all applications that store and manipulate data. In this guide, DBMS refers to the DBMS supported by Tracker—MS SQL Server, Sybase, Oracle, and MSDE.
dependent field	Dependent fields restrict available choices in user and choice fields. They are set up to impart process control and ease data entry during submit and update. Dependent fields are made up of a parent field and a child field to create a dependent field relationship.

dependent relationship type	A relationship between two issues in which the master role issue type and detail role issue types have some significant distinction, such as constraint of final state. For example, a parent issue can require all child issues to be closed before it can be closed.
detail-node role	The subordinate role of an issue in a relationship hierarchy. For example, in a parent-child relationship type, the detail-node role is Child.
distributed servers	Servers that contain different information. LDAP distributed servers can be configured to search for login information on more than one server. These servers use the information returned from any server to authenticate the login by means of referral .
duplicate relationship type	Duplicate relationships are dependent, non-nested relationships between two issues. Since any duplicate is typically a duplicate of any other duplicate that is related to the master-duplicate, duplicate relationships are configured to view peer relationships. This allows users to view all duplicates from the master-node as well as the detail-node. Initially the duplicate relationship type is setup to allow any issue type to be related to any other issue type.
field	A piece of information stored in an issue; for example: Title, Submit Date, and Resolution. The Server or Project Administrator can modify the labels for standard fields or customize the project database for your project by adding or modifying custom fields.
final state	A state that signifies that the particular issue is closed or complete. There may be more than one final state.
IDE	Integrated Development Environment. These are third-party software development products that incorporate compilers, debuggers, and so on in a single interface. Examples are Microsoft Visual Studio or Sybase PowerBuilder.

informational relationship type	Informational relationships are a simple informal relationship. Initially the informational relationship type is setup to allow any issue type to be related to any other issue type.
issue	A record in the Tracker database that contains information about an issue submitted by a Tracker user.
issue relationship	A relationship of issues grouped together and governed by a relationship type.
issue type	A classification that groups together related records so they can be managed differently from other related groups of records. All issues have an issue type, such as Change Request.
joins	Information stored in multiple tables combined in a single query. A join lets you combine rows logically across tables, producing a single output table.
LDAP	Lightweight Directory Access Protocol. An industry-wide standard protocol that is commonly used for user authentication and login.
log device	The database device where you will store log data. If you do not specify a log device, Tracker Administrator stores log data on the data device.
logging in	The process for entering Tracker and Tracker Administrator. Logging in involves entering a Tracker user ID, a Tracker password (if applicable), information about the DBMS you're logging in to, and optionally selecting a project to work on.
MAPI	Messaging Application Programming Interface. This type of mail system includes Microsoft Mail.
master-node role	Controls the attributes of the relationship. For example, in a parent-child relationship type, the master-node role is Parent.

Metrics	Metrics is a feature that collects statistics of your choosing on a given Tracker project, and can later present those statistics on a web page, as a graph or a table. By scheduling the collection of statistics at regular intervals, Metrics can help you keep track of incoming defect rates, fixed-versus-incoming rates, and so on.
mode	A system state in Tracker. There are eight such modes, or windows: Project, Who, Notes, User, User Group, Field, Form, and Terms. There is an icon for each one in the Tracker Administrator toolbar.
Notify	Notify is an optional component that automatically notifies users of issues via corporate e-mail. The product is used by Server Administrators only. It is VIM and MAPI compliant.
ODBC	An industry standard created by Microsoft that provides database users with a common way to access different databases. Each type of database requires a special ODBC driver to make it ODBC-compatible. ODBC drivers are provided with Tracker for each of the supported databases—MS SQL Server, Sybase, Oracle, and MSDE.
parent-child relationship type	A predefined relationship type used in issue relationships. The parent-child relationship type has a master role of Parent and a detail role of child. The parent-child relationship allows users to navigate from parent to child and child to parent and across relationships. The detail roles' final states constrain the master role's final state, such as ability to close. Parent-child relationships have closure constraints so that parent records cannot be closed until all of their children are closed.
PDIFF	PCMS Dimensions Interchange File Format (PDIFF) allows you to export your Tracker files to the Dimensions database. Please consult the Dimensions Administrator to set up the data fields before using this feature.
peer relationship	Two or more issues in relationships that have the same roles, such as two child issues of the same parent.

permission An option that controls whether a user can perform a particular Tracker action. For example, if you select the “Build Queries” permission for a user, that user is allowed to create new queries.

project A framework for grouping and managing data for a particular purpose, such as change requests for a software application. When you create a project, you give it a meaningful name that corresponds to the data you’re tracking, specify where information about the project is stored, and optionally customize aspects of the project, such as the users who are allowed to access it and the terminology used in the project’s submit and update forms.

Project Administrator One or more Tracker users who are members of the Administrator Group for the project. The Administrator Group is responsible for granting user permissions for the project. Project Administrators are typically managers or lead developers who need to be able to customize Tracker projects.

PVCSTRKUSR A set of tables that control database level permissions on MS SQL Server, Sybase, and Oracle. It is automatically created when the server is prepared for Tracker.

NOTE For more detailed information you can view the Tracker Database Schema Specification. Select Help | Search from the Help menu in Tracker Administrator and enter “Schema”.

query A search question that tells the database being queried what kind of information should be retrieved.

redundant servers Servers that contain the same information. LDAP redundant servers can be configured to allow logins to be processed when one or more servers are down.

referral In LDAP, one server directing a request to another LDAP server is called a referral.

relationship type	A set of rules governing how to make a relationship between issues. These rules may include role names, viewing rules, classification, allowed issue types, and closure constraints. Relationship types allow users to group and manage related issues.
repair	See Verify and Repair.
sa password	Stands for system administrator, and refers to the “sa” user in an MS SQL Server or Sybase environments. The sa user is the super user of the DBMS—this user has full privileges to the entire system.
SCR	A system change request. A record in the Tracker database that contains information about an issue submitted by a Tracker user.
server	For purposes of this guide, a server is a database where Tracker stores the basic tables (TRKMASTER tables) for a particular group of projects. Each Tracker project must be created on a server so it can have access to the TRKMASTER tables.
Server Administrator	The person who installs Tracker and Tracker Administrator is responsible for preparing servers and creating projects (unless a DBA is involved), creating Tracker users and Project Administrators, creating user groups, assigning permissions, and managing all Tracker projects on a particular server.
shared issue type	An issue type that is shared with a system level issue type for the purpose of creating relationships across projects.
single sign-on	An authentication protocol that allows a user to log in only once per session to use one or more Merant applications, as long as the user holds an active product license.
standard field	One in a standard list of fields that is provided in a project database by default at creation. Standard fields are central to the Tracker operation and may not be deleted. The Server

Administrator can only modify their labels. Some examples of standard fields are Title, Description, Owner, and Submitter.

system issue type An issue type created by a Server Administrator at the system level. A project administrator can connect issue types with system issue types so that issue relationships can be defined across projects.

system password Refers to the “system” user in an Oracle environment. The system user is the super user of the DBMS—this user has full privileges to the entire system.

TRKMASTER A collection of tables that exists on each server Tracker uses. It contains system-wide tables as a record of all projects on that server, and style sheets and pointers that control the Tracker user interface and allow it to relate projects to database records. These tables include:

- TRKCTL
- TRKPROC
- TRKGLB
- TRKREG
- TRKPRJ
- TRKSTY

NOTE For more detailed information you can view the *Tracker Database Schema Specification*. Select Help | Contents from the Help menu in Tracker Administrator and click Database Schema.

TrackerLink A component of Tracker used to access a subset of Tracker’s functionality so you can track issues from inside Version Manager, from the Tracker web client, or in an alternative IDE.

transition field Transition fields restrict field choices based on the value of the current field. Transition fields can only be choice fields.

user group A category that associates users with similar job descriptions and responsibilities. Submitter, Quality Engineer, and Support Engineer are typical *user groups*. You assign each user to a user

group when you add them to the Tracker database. User groups can be added or deleted, and users can belong to multiple user groups. All users automatically belong to the Public user group.

userspace

The same as “schema” in Oracle documentation. For more information, see Oracle’s *SQL Language Reference Manual*.

Verify and Repair

The tool used to ensure database integrity by checking for database corruptions and replacing lost or damaged data.

VIM

Vendor Independent Messaging system. This type of mail system includes Lotus Notes.

2 Implementing Tracker

In this Chapter

For information about . . .	See page . . .
How Tracker Fits Into Your DBMS Environment	38
People Involved In Managing a Tracker Project	39
Planning a Successful Tracker Implementation	44

How Tracker Fits Into Your DBMS Environment

Tracker uses a relational database management system, or DBMS, to store system change records, user configurations, queries, reports, and custom field definitions.

Refer to the readme for the supported versions of these databases:

- Oracle
- Microsoft SQL Server
- Sybase System
- MSDE (*for evaluation or training purposes only*)

NOTE MSDE is available free of charge; to use any of the other three databases, you must purchase it separately.

Tracker uses the DBMS you choose as a warehouse in which to store the information it needs for its tasks. Specific instructions are provided for specific DBMS tasks as needed.

Each DBMS stores Tracker data according to its own structure:

- Under Oracle, Tracker has its own database. The tables for *TRKMASTER* and each project are stored in a separate userspace within that database.
- Under MS SQL Server, MSDE, and Sybase, Tracker data exists on a server. *TRKMASTER* and each project are stored in a separate databases on the server.

For more information, see [Appendix A: “Disk Space Allocation” on page 293](#).

NOTE You can also use Tracker with more than one database type at a time if your environment requires it. You must point Tracker to the specific database at login time. See [“Logging In to Tracker Administrator” on page 56](#).

Before you begin using Tracker Administrator, we recommend that you familiarize yourself with basic tasks described in this chapter. If you have never used Tracker before, we recommend that you first install the product on a local workstation, as described in the *Installation Guide*, and then read the *User’s Guide* to learn how to use the product.

People Involved In Managing a Tracker Project

For best use, Tracker itself must be managed effectively. This section discusses the administration roles for Tracker.

Four Types of Administrators

There are potentially four types of administrators involved in a Tracker project:

- *Tracker Server Administrator*
- *Tracker Project Administrator(s)*
- *Tracker Web Client Administrator*
- *DBA (database administrator)*, sometimes called the System Administrator under SQL Server and Sybase

Of course, these roles are not always distributed among four different people. In some environments, there will be multiple servers with multiple administrators. In others, one or two people will perform all four roles. This guide makes these

distinctions to allow for the many scenarios possible within an organization.

Server and Project Administrators

Tracker functions are handled by the Server and Project Administrators. The Server Administrator performs Tracker-wide functions like installation, server preparation, and adding new projects, users, and user groups. The Project Administrators have authority to customize the way a specific project functions. The procedures in this guide are intended for both Server and Project Administrators, and are identified for you at the beginning of each section. See page [41](#) and [42](#) for a list of specific tasks.

Web Client Administrators

Tracker Web Client Administrators are responsible for configuring existing Tracker projects to be viewed via a web browser. Web client administration can be done completely remotely because the web client administration interface exists inside a web browser, too. Most of the project configuration is done beforehand by the Tracker Server Administrator. For more information, see [“How Tracker Fits Into Your DBMS Environment” on page 38](#).

Database Administrators

DBA is a role that pertains to the specific DBMS that runs in conjunction with Tracker—Microsoft SQL Server, Sybase, Oracle, or MSDE. The DBA will probably be involved in some Tracker installation, server preparation, and database maintenance, on an as needed basis. DBAs may find [Chapter 4, “Preparing the Server,” on page 61](#) and [Chapter 14, “Tracker Project Management and Recovery,” on page 271](#) to be of use to them.

Server Administrator Tasks

As the Server Administrator, you will:

- **Log in to Tracker Administrator.** You'll log in to Tracker Administrator as the Server Administrator for the database you'll be using to store Tracker information. This action prepares the database or database directory to receive a set of "master" tables that contain system-wide Tracker information.
- **Create projects.** Projects can be created from scratch, patterned after an existing Tracker project, or converted from a pre-8.x project.
- **Add users and assign permissions.** You'll add each Tracker user to the project by entering a user ID, assigning the user to one or more user groups, and optionally customizing permissions for user groups. You can choose to create additional user groups if the default set is inadequate for your organization. Permissions control which Tracker operations group members can perform.
- **Customize fields, forms, and terms.** You can custom-design aspects of a project to suit your particular development environment. For example, instead of the term "system change request," you could substitute the term "software defect," "customer issue," or any term you choose.
- **Integrate Tracker with Version Manager.** You can customize what types of Version Manager information are made available in issues for the project. It also allows you to assign the Version Manager configuration files Tracker will use.
- **Manage projects, users, and user groups.** When you're ready to create additional projects, you can copy one of the projects, or simply copy users, user groups, and permissions that are relevant to the new project.

- **Handle database maintenance and recovery.** You'll perform database backups and recoveries, table re-indexing, and maintain database integrity for all Tracker projects on your server.

Project Administrator Tasks

As a Project Administrator, you will:

- **Log in to Tracker Administrator.** You'll log in to Tracker Administrator and select the project you want to work on.
- **Customize users on a project-by-project basis.** You can copy selected users from one project to the next to determine who can access specific projects.
- **Customize user group permissions on a project-by-project basis.** You can copy user group permissions between projects and then customize them.
- **Customize fields, forms, and terms on a project-by-project basis.** You can modify fields, forms, and terminology, and create custom fields for each individual project.
- **Create relationships and issue types.** You can create issue types to uniquely identify the different types of issues users submit into Tracker. You can also define relationships to put these issue types into a hierarchy or relate specific ones together.
- **Set up users' notification options.** You can specify conditions, or "rules," under which a specific user will be notified automatically about a specific type of event through the Tracker In Tray or e-mail (if Notify has been installed).
- **Do periodic checks on database integrity.** You'll use the Verify and Repair tool to verify that your project database is intact and not corrupted. If your Server Administrator has given you

the necessary permissions, you may also repair project database corruptions if necessary.



Sample Scenario

Brad wanted his QA system to contain all the valid information available in his old system. So, in preparation for setting up the company's defect tracking function on Tracker, Brad queried his Excel spreadsheet and gathered the following information:

- *A report of all active projects*
- *A report of all users for each project*
- *A description of the information each user can access on a project*
- *A list of all project managers*

Brad found that there were 12 current development projects and 120 users, five of whom were project managers. Brad's analysis further discovered that out of the five project managers, two project managers were each responsible for three projects, one project manager was responsible for one project, and two project managers each were responsible for two projects. Brad decided that these five project managers would best fit into the Tracker workflow as Project Administrators. He requested five additional copies of the Merant Tracker Administrator's Guide from Merant so that each Administrator would have their own copy.

Brad informed his boss that "Given the size of our QA effort, incorporating the power of Tracker will take a little planning. I'll call a meeting of the soon-to-be Tracker Project Administrators to make sure they are on board for this change in workflow."

“Just do what it takes to get the new system in place. And don’t waste a lot of time in meetings. How are you going to keep this discussion on track?” the boss asked.

Fortunately, Brad had an answer. The best way to get the meeting started would be to use the Tracker Implementation Checklist (see page 47) as an agenda. He also invited Denise, the DBA assigned to Tracker, to keep her up to speed on the activities and needs of the new user group that would be accessing her database. (See page 71 for a continuation of this scenario.)

Planning a Successful Tracker Implementation

Tracker’s simplicity belies a tremendous capability. To use this capability effectively in the long term as well as the short term, it’s important to plan how you will use Tracker in your own particular organization.

Size of Your Organization

How you fill the roles of DBA, Server Administrator, and Project Administrator depends on the size of your organization.

Although the organizational structures in which Tracker can be used vary, they typically fall into three classifications:

- Small shops (fewer than 25 Tracker users) may combine the roles of database administrator (DBA), Server Administrator, and Project Administrator.
- Medium-sized shops (26 to 100 Tracker users) may have a combined DBA/Server Administrator role, but have separate Project Administrators.

- Large shops (more than 100 Tracker users) may have separate DBAs, Server Administrators, Project Administrators—and may even have multiple instances of these roles on various database types (SQL Server, Sybase, and Oracle) throughout the organization.

All three roles—DBA, Server Administrator, and Project Administrator— must be filled regardless of the size of your organization. The [“Tracker Implementation Planning Checklists” on page 47](#) help you see which role is responsible for which part of Tracker’s implementation and maintenance, as well as highlight the issues you need to address in your plan.

Tracker’s Impact on the QA Environment

Every defect tracking environment is affected by:

- Corporate level goals and culture such as:
 - How the corporation defines successful defect tracking.
 - Team versus independent work styles.
- Technological constraints such as:
 - Whether the project team members are on networks.
 - Whether each project team member has their own PC.
- Human idiosyncrasies—communication issues such as:
 - How confidential the organization considers defect tracking information.
 - Team member concerns about how their duties and autonomy are going to be affected by the new system.

Recognizing how these elements operate in your defect tracking environment helps you implement Tracker more effectively.

We recommend you write a brief implementation plan that includes the items in the following checklists.

Tracker Implementation Planning Checklists

The following checklists are designed to help streamline the process of setting Tracker up in your environment in a manner that will best lend itself to easy maintenance. There are five checklists:

This checklist ...	Is on page ...
Why Are You Implementing Tracker?	48
What are the Organizational Goals in Implementing Tracker?	49
How Will You Avoid Stumbling Blocks?	50
Who Will Do What?	52
What Existing System Information Can You Gather?	53

Why Are You Implementing Tracker?

Knowing why you’re implementing Tracker will help you plan and define goals and stay on track as you move into the maintenance phase. Answering the following questions will help you set permissions and user groups up to more effectively realize your vision.

Table 2-1. Implementation Focus Checklist

Implementation Issue	Which Administrator is best suited to address the issue?		
	Database	Server	Project
What current defect tracking problems are you trying to solve (example: no single source of issue information)?			
What will you be using Tracker for? <ul style="list-style-type: none">■ Problem tracking with development■ Problem tracking across the organization■ Product tracking			
What is Tracker’s focus?			
Who will benefit from the data Tracker tracks and generates?			
How will the system be managed?			
What will the communication structure be between users and administrators? Among administrators?			
What degree of autonomy will be allowed in reporting change requests?			

Table 2-1. Implementation Focus Checklist *(cont.)*

Implementation Issue	Which Administrator is best suited to address the issue?		
	Database	Server	Project
What will the communication structure be between users and administrators? Among administrators?			
How will implementing Tracker affect the way users interact with customers (example: field test sites)?			

What are the Organizational Goals in Implementing Tracker?

You will be able to implement Tracker more effectively if you define and create plans for attaining the following goals.

Table 2-2. Organizational Goals Checklist

Organizational goals	Which Administrator is best suited to address the issue?		
	Database	Server	Project
What are the corporate issue-related goals?			
What are the areas for improvement in the current system?			
What is the vision of a successful QA function?			
What quality goals do you want to achieve?			

How Will You Avoid Stumbling Blocks?

Certain issues should be handled during the planning phase of the implementation, before they cause problems during implementation.

Table 2-3. Risk Management Checklist

Possible issues	Which Administrator is best suited to address the issue?		
	Database	Server	Project
Do all the users have a PC?			
Are all the users on the network?			
Which server will the project information be stored on? Is there enough space?			
Does your organization need automatic e-mail notification via Notify? And, if so, is your e-mail system VIM, MAPI, or SMTP compliant?			
Who is responsible for database problems/modifications?			
What additional duties may be added to each user's task list?			
What degree of autonomy will be allowed in reporting change requests?			
What is the issue life cycle?			
Which groups in your organization have what roles in the issue life cycle?			
What rights (permissions) do they need to accomplish their tasks?			

Table 2-3. Risk Management Checklist *(cont.)*

Possible issues	Which Administrator is best suited to address the issue?		
	Database	Server	Project
What routine updates will be required in the new issue life cycle?			
How will exceptions to the issue life cycle be handled—Exception flows or returns and special cases.			
What kind of information do you want to track? <ul style="list-style-type: none"> ■ Custom queries ■ Custom reports ■ Field support ■ Form support 			
What kind of metrics do you want to generate?			
How will you use the available information? <ul style="list-style-type: none"> ■ Metrics ■ Public vs. private reporting ■ Team/group evaluation ■ Individual evaluation 			
How will the users be trained on: <ul style="list-style-type: none"> ■ Tracker ■ Your organization's implementation of Tracker 			

Who Will Do What?

Defining roles helps prevent problems with data integrity.

Table 2-4. Role Definition Checklist

Who will:	Which Administrator is best suited to address the issue?		
	Database	Server	Project
Add new users to projects?			
Add new custom fields to projects?			
Create new user groups? <ul style="list-style-type: none">■ Maintain synch with goals■ Maintain synch with focus■ Licensing limitations■ System access available for all affected users?			
Create new databases?			N/A
Assign permissions to user groups?			
Copy users?			
Copy user groups?			
Copy custom fields?			
Handle database recovery?			

What Existing System Information Can You Gather?

If you gather the following information in advance, the second phase of the implementation will go more smoothly. Ultimately, the Server Administrator will need to have all of this information.

Table 2-5. System Information Checklist

Information to gather:	Which Administrator is best suited to gather the information?		
	Database	Server	Project
Database server names			
Database user IDs Note: This is only necessary if you are using SQL Server, Sybase, or Oracle as your database.			
Tracker user IDs and associated information: <ul style="list-style-type: none"> ■ Full name ■ E-mail address ■ Telephone 			
Projects each user needs access to			
Permissions each user must have to do his or her job			
Custom organization-wide user groups to be created and the specific permissions to be allowed			

Table 2-5. System Information Checklist

Information to gather:	Which Administrator is best suited to gather the information?		
	Database	Server	Project
Customer project-specific user groups to be created and the specific permissions to be allowed			
The name and user ID of at least one Project Administrator per project			

3 Logging In to Tracker Administrator

In this Chapter

For information about . . .	See page . . .
Overview	56
Logging In to Tracker Administrator	56
Practicing with the Learning Projects	58
Switching to Another Project	59
Logging Out of Tracker Administrator	59

Overview

Who Should Read
This Chapter?

This chapter is intended for Server Administrators *and* Project Administrators.

Why Read This
Chapter?

In this chapter, you'll learn how Tracker connects with each of the supported databases and how to get started with the program. After you finish this chapter, Server Administrators will be ready to start creating projects; Project Administrators will be ready to begin customizing their projects.

NOTE Many of the terms used in this chapter are defined in the ["People Involved In Managing a Tracker Project" on page 39](#).

Logging In to Tracker Administrator

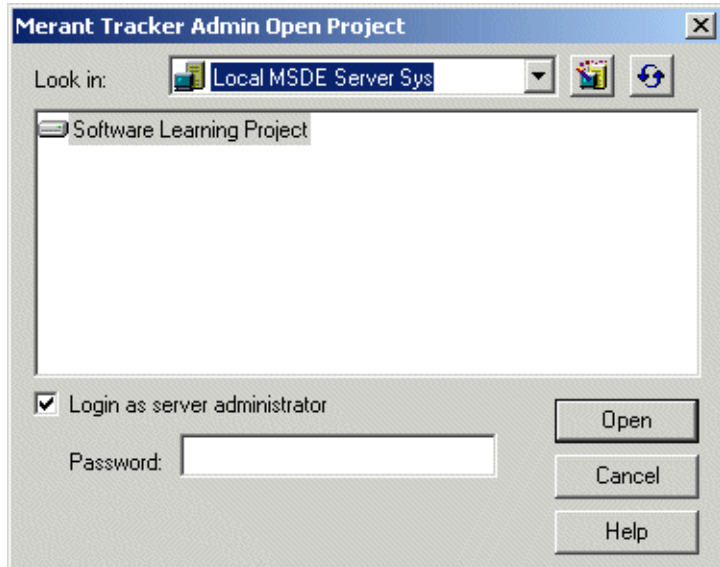
To use Tracker Administrator, you must log in as either the Server Administrator or a Project Administrator. There is only one Server Administrator for each server. There can be multiple Project Administrators—one or more for each project. Project Administrators are set up by the Server Administrator. For setup instructions, see ["Setting Up Project Administrators" on page 116](#).

This procedure explains how to log in as the Server Administrator or Project Administrator.

NOTE Use this procedure only after Tracker has been set up on the server and the server has been prepared. See [Chapter 4, "Preparing the Server," on page 61](#) for more information.

To log in:

- 1 Start Tracker Administrator. The Merant Tracker Admin Open Project dialog box appears.



The default is to log in as server administrator; if you clear this check box, you will be logged in as project administrator. If this is the first time you have logged in, or if the server administrator password has not yet been set, leave the password field blank.

- 2 If you do not see the projects you want to work on, try selecting a different server from the server drop-down list.



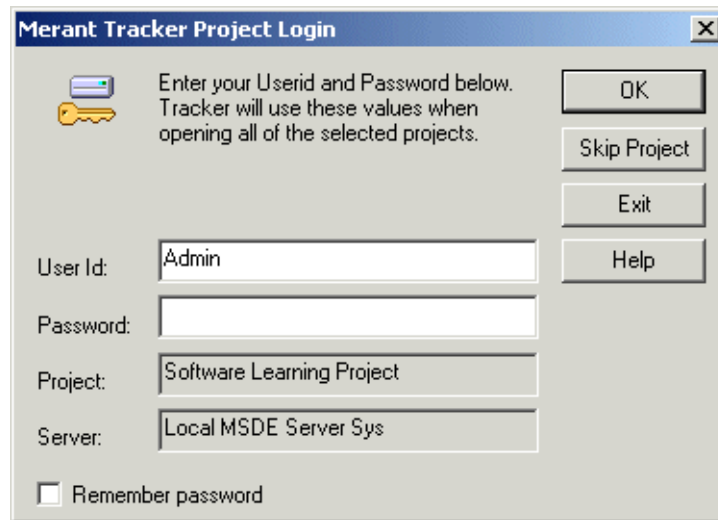
You can also try clicking the Refresh button



to update the list of projects.

- 3 Select the project or projects you want to work on; then click Open. The Merant Tracker Project Login dialog box appears.

NOTE This dialog does not appear if you are logging in as the system administrator.



- 4 Enter your User ID and password; then click OK.

Practicing with the Learning Projects

You can practice using Tracker Administrator (and Tracker) with the Learning Projects. The Learning Projects come with your Tracker application and contain sample data that you can practice on until you are comfortable enough to work on your own projects. When you log in, just accept the defaults in the Merant Tracker Administrator Login dialog box. Tracker Administrator automatically opens the Software Learning Project.

NOTE The Learning Projects are MSDE-based projects. If you want to use another DBMS, you can create a dummy SQL Server, Sybase, or Oracle project by copying one of the Learning Projects. For instructions, see [“Creating a New Project Based on an Existing One” on page 100](#).

Switching to Another Project

You can switch to projects on a different server without exiting Tracker Administrator.

To switch to a different server:

- 1 Select File | Switch Server.
- 2 Use the Merant Tracker Admin Open Project dialog box to log in to a different server. See [“Logging In to Tracker Administrator” on page 56](#).

Logging Out of Tracker Administrator

- To log out without exiting Tracker Administrator, select File | Logout.
- To log out of all projects and exit Tracker Administrator, do one of the following:
 - Double-click the system menu symbol at the top left corner of the main window.
 - Select File | Exit.

4 Preparing the Server

In this Chapter

<i>For information about . . .</i>	<i>See page . . .</i>
Overview	62
Elements of Server Preparation	63
About Tracker Databases	64
About Devices, Tablespaces, and Datasources	64
About the Tracker-DBMS Login Relationship	66
Information You Need Before You Begin	68
Working with Server Definitions	72
Preparing the Server	79

Overview

Who Should Read This Chapter?

This chapter is for the person who will be creating and setting up the Tracker server. In some cases, the Tracker Server Administrator will have full privileges on the DBMS that Tracker is using to store its data, and can do the whole process. In other cases, the Database Administrator, or DBA (the person who administers the DBMS), will have to perform some tasks. The procedures in this chapter are divided with this in mind—one section for server preparation by the Server Administrator, and another for server preparation where the DBA is involved. You choose which section you wish to follow.

Why Read This Chapter?

In this chapter, you will learn how to *prepare the server* for Tracker. When you are finished with this chapter, you will be ready to create projects in [Chapter 5, “Creating and Managing Projects,”](#) on page 85.

You cannot prepare the server until you have:

- Created the devices under the DBMS for Sybase and created the datasources under the DBMS for SQL Server.
- Installed Tracker successfully. Under Microsoft SQL Server, Sybase, and Oracle, this means that both the server *and* the client portions of Tracker and the DBMS have been installed.

This chapter assumes you have already installed Tracker successfully (*Merant Tracker Installation Guide*) and that you know how to log in to Tracker Administrator ([Chapter 3](#)).

Elements of Server Preparation

Preparing the server involves:

- 1 Pointing Tracker to the proper data and log devices and allocating disk space for the Tracker database server.
- 2 Creating the Tracker master database (or userspace in Oracle) called TRKMASTER, which contains the master tables that all Tracker projects need to function.
- 3 Creating user IDs for administrators, in Tracker and possibly the DBMS.
- 4 Creating user IDs for Tracker users, in both Tracker and the DBMS.
- 5 Establishing a Tracker-DBMS login relationship for the Tracker server and project administrators.
- 6 Establishing a Tracker-DBMS login relationship for the Tracker users.

Tracker attempts to do as many of these tasks as it can. How much Tracker can do depends on which DBMS you are using and whether you supply the DBMS's *sa* or *system* password. If you do not, you will have to do a few additional tasks by hand.

Under Sybase, your DBA has to create devices, but if you supply the *sa* password, Tracker can do the rest.

Under Oracle, your DBA has to create tablespaces and the database for Tracker. If you supply the *system* password, Tracker can do the rest.

About Tracker Databases

Tracker creates a set of master tables called TRKMASTER that contain information about all the projects on the server. The way it stores TRKMASTER and the project tables depends on which DBMS you are using.

Under ...	Tracker stores the tables for TRKMASTER ...
SQL Server and Sybase	and each project in separate databases on the server.
Oracle	and each project in tablespaces within the same database on the server.

About Devices, Tablespaces, and Datasources

Create your devices, tablespaces, or datasources *before* you install Tracker and attempt to prepare the server. Devices, tablespaces, and datasources are the areas set aside on a server for storing specific data. For the purpose of this manual, the terms are equivalent, except that “device” is the term used in Sybase environments, “tablespace” is the term used in Oracle environments, and “datasources” is the term used by SQL Server.

Under Sybase, we recommend you create separate devices and datasources for storing your project data and transaction logs. This improves performance and data recovery, reduces the risk of damaged data, devices, or datasources, and allows you to control the size of the transaction log, which could get inordinately large if you keep it with the project data.

Another point to consider is this: If both data and log devices are on the same drive, and the drive fails, then both project and transaction data will be lost.

NOTE Never use the SQL Server or Sybase master database (MASTER.DAT) for storing Tracker data. This database contains information internal to SQL Server and Sybase. If you do use it, and your database becomes unstable or corrupt, your entire SQL Server or Sybase database could become corrupt.

Creating Devices and Tablespaces

MS SQL Server 7.0

The database datasources are created during the process described in [“Preparing the Server” on page 79](#).

Sybase

To create a database device in Sybase:

Using your Sybase server tools, create a device for the Tracker Master and project tables. Use the disk init command. This command requires several parameters:

```
name = logical name_of_device,
physname = os_location_of_device,
vdevno = virtual_device_number
size = size_of_device_in_pages
[, vstart = virtual_address, cntrltype=
controller_number]
[, contiguous]
```

Consult your Sybase documentation for more information.

Oracle

You must create a database and a tablespace in the Oracle server before connecting to Tracker. Consult your Oracle documentation for instructions.

About the Tracker-DBMS Login Relationship

When you login to Tracker and Tracker Administrator, a relationship must be established between the Tracker and DBMS login so that information can be exchanged between Tracker and the DBMS. This relationship is established by supplying two user IDs (and passwords) when you login—one for Tracker and another for the DBMS.

For Tracker Users

The Tracker-DBMS login for Tracker users can be set up in two ways:

- A one-to-many relationship
- A one-to-one relationship.

Which one you choose depends on your specific needs and how involved your DBA wants to be in Tracker administration.

One-to-Many (the Default Login Method)

The *default login method* is a one-to-many relationship. Each user has a unique Tracker user ID, but uses a common DBMS user ID. Therefore, for a single DBMS user ID, there are many Tracker user IDs. This setup minimizes DBA involvement in Tracker

administration, since only one user ID has to be created on the DBMS.

For example

Each Tracker user...	...uses the same DBMS user ID.
<i>jdoe</i>	<i>pvcstracker</i>
<i>amcgill</i>	<i>pvcstracker</i>

One-to-One (the Multiple Login Method)

The *multiple login method* is a one-to-one relationship. Each Tracker user ID has a corresponding, unique DBMS user ID. For every DBMS user ID, there is one Tracker user ID. Since many user IDs are created on the DBMS, the DBA will have greater involvement in this type of setup. (The DBMS will have multiple user IDs for Tracker users—hence the name *multiple login method*.)

For example

Each Tracker user...	...has another user ID for the DBMS.
<i>jdoe</i>	<i>jdoe</i>
<i>amcgill</i>	<i>amcgill</i>

For Tracker Administrators

The Tracker-DBMS relationship for Server Administrators can take one of two forms:

- The *Admin* Tracker user ID plus the *sa* or *system* DBMS user ID, or
- The *Admin* Tracker user ID plus a *trkadmin* DBMS user ID that has ownership of Tracker data.

Using the sa or system Login

In this method, the Tracker Server Administrator has been given the DBMS *sa* or *system* password and uses it for all Tracker-DBMS interactions. This means, of course, that the Server Administrator will have all the same permissions as the DBA.

Using the Admin Login

In this method, the DBA has provided the Tracker Server Administrator with a special *trkadmin* user ID and password on the DBMS. Of course you can use any name you wish.

This method is usually used when the DBA does not want to give out the *sa* or *system* password. Specific DBMS permissions must be set up for the *trkadmin* user to make it possible to perform Tracker administrator functions. See your DBMS documentation for specific information on permissions.

Information You Need Before You Begin

Before you begin preparing the server, make sure you have all the information you need. It will help if you answer the following questions:

1 *Has the Tracker application been installed yet?*

The Tracker application must be installed before you attempt to prepare the server. This means both the server *and* the client portion of the installation. After you prepare the server, each workstation that connects with Tracker must be configured for Tracker and have the required DBMS components installed. If this is not done, see the *Merant Tracker Installation Guide*.

- 2 *Have the devices, tablespaces, or datasources been created yet?*

The Tracker data and log devices must be created before you prepare the server. If they have not, see “[About Devices, Tablespaces, and Datasources](#)” on page 64.

- 3 *On which devices do you want to place Tracker data and log information?*

We recommend using separate data and log devices. See page 64 for more information.

- 4 *How much space do I want to allocate for data and log devices?*

See our space recommendations on page [293](#).

- 5 *Will the Tracker Server Administrator be allowed full access to the DBMS?*

In SQL Server and Sybase, this means they will have the *sa* password. In Oracle, this means they will have the *system* password.

If the Server Administrator...

Then the...

Has full access

Job will be fairly simple because using the *sa* or *system* password gives Tracker the power to do most functions for you.

Does not have full access

DBA will need to be involved in server preparation, current and future project creations, and will need to provide a *trkadmin* user ID with special privileges for the Server Administrator.

- 6 *Do you want your Tracker users to use a common DBMS login or would you prefer they used unique ones?*

In making this decision consider:

- If users have a common login, there will be only one user ID for users (*pvcstracker*). Tracker will create this ID when the *trkmaster* tables are created. If users have unique logins, the DBA will have to create new user IDs whenever there are new Tracker users.
- If users have a common login, your DBMS tools will not be able to distinguish between users. So you will not be able to tell which login session belongs to which user. They will all be called *pvcstracker*. However, you can use Tracker's Who function to detect the person associated with each session. If users have unique logins, your DBMS tools will be able to view exactly who is logged on at any time.

See [“About the Tracker-DBMS Login Relationship”](#) on page 66 for more information.

If you, as the Server Administrator, have full DBMS privileges (meaning you have the *sa* or *system* password), Tracker can help you do most or all functions involved in preparing your server. [“Preparing the Server” on page 79](#) leads you through the procedure step by step.



Sample Scenario

Using the implementation planning checklists (see [page 47](#)), the new Tracker Administrators laid out the organizational structure and workflow for their new QA system in record time. The day came to set up all the ongoing QA projects and active users in Tracker. Brad called Denise to verify that SQL Server was ready.

“Yes,” she said. The password is wexler.”

“That’s the first password in the dialog box?”

“No, that’s your admin password. I’m talking about the DBMS password.”

Brad started Tracker Administrator and looked at the Login dialog box.

“Oh, I see it here. Thanks, Denise! I should be okay on my own, but I’ll call you if I need help,” said Brad.

Brad hung up and began to fill in the appropriate fields in the dialog box. (See [page 82](#) for a continuation of this scenario.)

Working with Server Definitions

Once the database server has been set up, the Administrator should prepare server definitions for the end-users. This allows them to simply select a server from a drop-down list, and connect to all of their projects from there.

The server definitions contain information that in older versions of Tracker was known as connection information. This information appeared under the Advanced button of the Login dialog. Now that information is captured in server definitions.

To create, edit, or delete server definitions, or to convert connection information from older versions of Tracker to server definition format, use the Tracker Server Wizard.


The Tracker Server Wizard

The Tracker Server Wizard helps you create server definitions, edit server definitions, and convert or capture connection information such as DBMS user ID and DBMS passwords. The Tracker Server Wizard is available in either Tracker Administrator or the Tracker Windows end-user application from a button on the Open Project dialog box.

Converting Old Server Definitions

In versions of Tracker before 7.x, server information was stored in configuration files (such as PVCSTRK.INI). Before those projects can be used with Tracker 7.x or higher, you need to convert the server information to a Tracker server definition. The Tracker Server Wizard helps you do this.

To reach the Tracker Server Definition Wizard and convert old server information into a new server definition:


- 1 Choose File | Login. The Merant Tracker Admin Open Project dialog box appears.
- 2 Click the Tracker Server Wizard button . The first screen of the Tracker Server Wizard appears.
- 3 Click the radio button labeled, **Convert old server definition(s)**, then click Next. A screen appears asking for the name of the configuration file you want to convert. By default, the Tracker Server Wizard will select PVCSTRK.INI, but you can select a different file using the Browse button. If PVCSTRK.INI is not the correct file, browse for or enter the name of the correct file.
- 4 From the list of servers that appear in the list box, select the server whose definition you want to convert; then click Next. The Tracker Server Wizard displays a confirmation screen indicating that it has enough information to convert the server definition, and shows you the information it has so far. Click the Back button to return to a previous screen; otherwise, click Finish to complete the server definition conversion.
- 5 The Merant Tracker Open Project dialog box appears, with the converted server showing in the drop-down list box.

Creating New Server Definitions

You use the Tracker server definition wizard to create a new server definition. This allows you to create and access projects on that server.

To reach the Tracker server definition wizard and create a new server definition:

- 1 Before starting, you should have the following information:

- The DBMS type and location
 - Your DBMS user ID and login
- 2 Choose File | Login. The Merant Tracker Admin Open Project dialog box appears.
 - 3 Click the Tracker Server Wizard button . The first screen of the Tracker Server Wizard appears.
 - 4 Click Next to accept the default option **Create a new server definition**. A screen appears asking for the name of the server, as well as the DBMS type and the DBMS location.
 - 5 Enter the name you want to give the server, and the DBMS type and location; then click Next.

NOTE If the directory you specify as the DBMS location does not exist, the Tracker Server Wizard will not create it for you.


- 6 When you create a server definition, a screen appears asking for your DBMS user information (user ID and password). If you are already a Tracker user, the Tracker Server Wizard will attempt to use your default user ID and password; otherwise, enter the information into the appropriate fields.

- 7 If you are creating a server definition for a user with Administrator privileges, select the **DBMS Administrator Information** check box and enter the Administrator user ID and password.
- 8 Select the **Remember passwords** check box marked to have Tracker retain your password.

NOTE If you do not have a server prepared for use with Tracker, you will see a dialog box asking if you would like to prepare a server now. For details, see [“Preparing the Server” on page 79](#).

- 9 The Tracker Server Wizard displays a confirmation screen indicating that it has enough information to create the server definition, and shows you the information you have entered so far. If any of the information is incorrect, click the Back button to return to a previous screen; otherwise, click Finish to complete the server definition.
- 10 The Merant Tracker Open Project dialog box appears, with the new server showing in the drop-down list box.

Troubleshooting

If you see the error “Cannot connect to the selected server. Hit refresh to reconnect”, try clicking the Refresh button  to update the list of servers.


If the new server still does not appear, the Tracker server definition wizard has not recognized the server. Check with your database administrator to be certain the database server is working properly.

Editing Server Definitions

You can use the Tracker server definition wizard to edit an existing server definition. You might want to do this because:

- You want to change the name of the server
- You need to change the DBMS user ID or password


To edit the Tracker server definition wizard server definition:

- 1 Choose File | Login. The Merant Tracker Admin Open Project dialog box appears.
- 2 Click the Tracker Server Wizard button . The first screen of the Tracker Server Wizard appears.
- 3 Click the **Edit a server definition** option. A screen appears showing a list of servers.
- 4 Select the server definition to edit; then click Next. A screen appears showing the server information.
- 5 Change the server information as desired; then click Next.
- 6 The Tracker Server Wizard displays a confirmation screen indicating that it has enough information to complete the server definition, and shows you the information you have entered so far. If any of the information is incorrect, click the Back button to return to a previous screen; otherwise, click Finish to finish editing the server definition.
- 7 Tracker connects you to the selected server, and the Merant Tracker Open Project dialog box appears.

Deleting Server Definitions

You can use the Tracker server definition wizard to delete an existing server definition.

To reach the Tracker server definition wizard and delete a server definition:

- 1 Choose File | Login. The Merant Tracker Admin Open Project dialog box appears.
- 2 Click the Tracker Server Wizard button . The first screen of the Tracker Server Wizard appears.
- 3 Click the **Delete a server definition** option. A screen appears showing a list of servers.
- 4 Select the a server definition to delete; then click Next. The Tracker Server Wizard displays a confirmation screen showing the information for the server you wish to delete.
- 5 If you have selected the wrong server, click the Back button to return to a previous screen; otherwise, click Finish to finish deleting the server definition.

Tracker deletes the server definition, and connects you to the next server in the drop-down list on the Merant Tracker Open Project dialog box.

Making Server Definitions Available to All Users

The Tracker Administrator can provide server definitions to all users without exposing the DBMS login information by placing the server definitions into the CONNECT.INI file. Local server definitions are stored in an encrypted format in teach user's PVCSTRK.INI file. By using an editor such as Notepad, the Administrator can copy the server definitions from his PVCSTRK.INI file to the CONNECT.INI file.

The server definitions are marked in the PVCSTRK.INI file with the following heading:

```
[Server Definitions]
```

After copying the server definitions section to the `CONNECT.INI` file, the Administrator can ensure that all users see the `CONNECT.INI` file by making sure that each user's `PVCSTRK.INI` file contains the following entry:

```
[Tracker]  
sys=v:\tracker\sys
```

where V is a drive on the network or is otherwise available to all users.

Preparing the Server

This procedure is for Tracker Server Administrators with full access to the DBMS. You have to know the DBMS *sa* or *system* password. Even though the Server Administrator will prepare the server, the DBA still has to create the devices.

For the DBA

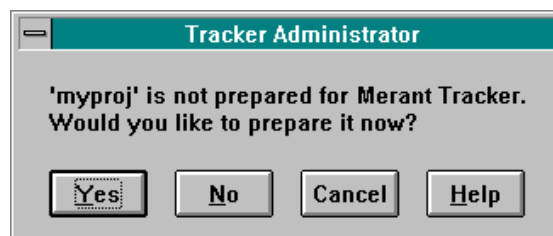
- 1 Under Sybase, create the data and log devices using your DBMS tools. Consult your DBMS documentation for specific instructions, or see ["Creating Devices and Tablespaces" on page 65](#) for information.

Under Oracle, create the tablespaces and the Tracker database using your Oracle tools. For instructions, consult your Oracle documentation.

For the Server Administrator

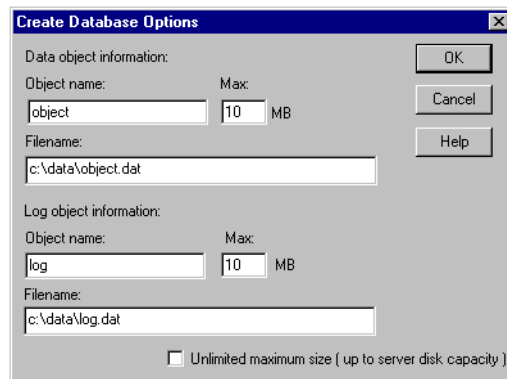
NOTE If the Tracker Server Administrator does not have full database permissions, the DBA will need to do this entire procedure.

- 2 Login to Tracker Administrator as Server Administrator using steps 1 - 4 beginning on page 57. When you click OK, Tracker displays:



- 3 Click Yes. Then proceed to step 4, 5, or 6, depending on which database you are using.

- 4 For SQL Server 7.0:
 - a Enter the name and pathname for the data object you wish to create.
 - b Set the maximum size, in Mb, for the data object, or, check the “Unlimited maximum size” check box to allow the data object to grow up to the limit of available free space. Selecting “Unlimited” will cause the “Max” fields to disappear. (See [page 293](#) for specific recommendations on disk space.)
 - c Follow the same process for the log object. (Note that it is a good idea to keep the data and log objects on separate physical devices.)



5 For Sybase:

- a Select the data device name and enter the amount of disk space you want to allocate. (See [page 293](#) for more information.)
- b Select the log device name and enter the amount of disk space you are going to allocate.

The dialog box is titled "Create Database Options". It has two sections. The first section is for the Data Device, with a dropdown menu showing "datadev1", a "Size" field with "25" and "MB", and "Space Available: 37 MB". The second section is for the Log Device, with a dropdown menu showing "trklog", a "Size" field with "7" and "MB", and "Space Available: 6 MB". There are "OK", "Cancel", and "Help" buttons on the right.

6 For Oracle:

- a Select the *default tablespace* name. Enter a quota allocation if you want, but it is not necessary. If you don't, quota usage will be unlimited. Note that temporary tablespace is not used in this Tracker release. Just select TEMP in this field.

The dialog box is titled "Create Userspace Options". It has two sections. The first section is for the Default Tablespace, with a dropdown menu showing "TRACKER", a "Quota" section with a checked "Unlimited" box and a "4" MB field, and "OK", "Cancel", and "Help" buttons. The second section is for the Temporary Tablespace, with a dropdown menu showing "TEMP", a "Quota" section with a checked "Unlimited" box and a "2" MB field.

- b Click OK.

Tracker creates the:

- TRKMASTER database under SQL Server and Sybase, or the TRKMASTER userspace under Oracle,
- TRKMASTER tables, placing them in the TRKMASTER database or userspace,
- *Admin* user in Tracker (the Server Administrator login),

- *pvcstracker* user for the DBMS (the users' default DBMS login). The password will be *Tracker*.
- *pvcstrkusr* role or group



Sample Scenario

Using the implementation planning checklists (see page 47), the new Tracker Administrators laid out the organizational structure and workflow for their new QA system in record time. The day came to set up all the ongoing QA projects and activate users in the Tracker. Brad called Denise to verify that SQL Server was ready.

"Yes," she said.

"Thanks, Denise, I'll create a server definition so I can start creating projects."

Brad started Tracker Administrator and looked at the Open Project dialog box. The MSDE server was listed, but Brad needed to create the real-work server, so he clicked the Tracker Server Wizard icon. The Server Wizard appeared and Brad selected Create a new server definition.

Brad entered the name for the server, selected "Tracker SQL Server" as the DBMS Type, and entered BBSD3 as the DBMS location.

At the next screen, Brad entered the DBMS user ID and password that he and Denise had agreed on, and then, because he was going to set up projects, started to enter the DBMS Administrator user ID and password. He called Denise again.

"Hey, Denise, what's the--"

"The login is 'sa' and the password is 'wexler'."

"Thanks, Denise. I should be okay now."

Brad hung up, filled the DBMS Administrator user ID and password, and the new server appeared in the Open Project dialog box's drop-down list.

See page [88](#) for a continuation of this scenario.

5 Creating and Managing Projects

In this Chapter

For information about . . .	See page . . .
Overview	86
Elements of Project Creation	87
About Projects	87
Creating a Project	89
Upgrading Your Projects	92
Reattaching an Existing Database	98
Creating a New Project Based on an Existing One	100
Renaming an Existing Project	102
Deleting a Project	103

Overview

Who Should Read This Chapter?	This chapter is for the Server Administrator, who is the person responsible for creating and maintaining Tracker projects. DBAs may also be interested in this chapter, especially "Reattaching an Existing Database," on page 98.
Why Read This Chapter?	<p>In this chapter you will learn how to create new Tracker projects, copy existing project environments, convert older projects, reattach projects to Tracker, and rename and delete projects. When you are finished with this chapter, your project will be ready for adding users in Chapter 6, "Creating and Managing Users," on page 105.</p> <p>This chapter assumes you have already installed Tracker successfully (<i>Merant Tracker Installation Guide</i>), that you know how to log in to Tracker Administrator (Chapter 3, "Logging In to Tracker Administrator," on page 55), and that you have already prepared your server (Chapter 4, "Preparing the Server," on page 61). You cannot create projects if you have not prepared your server.</p>

Elements of Project Creation

There are three components to creating a project:

- 1 Adding the project to Tracker
- 2 Specifying the devices and space allocations for the project
- 3 Creating the project database

To create a project, you must have full DBMS privileges (the *sa* or *system* password).

About Projects

A *project* is a way of managing information related to a specific development effort. It contains all the information about issues you want to track, and helps you manage them. Projects also include information about the users and user groups who will access those particular data records, as well as any customizations you make to the work environment.

Besides the master database, TRKMASTER, there is one additional database per project. Tracker keeps a list of all of these project databases inside TRKMASTER.

Tracker databases take a different form, depending on which DBMS you are using:

Under this DBMS ... Tracker creates ...

MS SQL Server,
MSDE, and Sybase

A separate database for each new project. Within each project database, it stores a set of tables that contain information for that specific project.

Oracle

A new userspace for each new project. In this userspace, Tracker stores a set of tables that contain information for that specific project.

Devices and Tablespaces

For each project database, you allocate space for project and log data on the devices or tablespaces you created. (See [Chapter 4, “Preparing the Server,” on page 61](#).) To determine how much space you need, see the guidelines for disk space allocation on page 293.

You cannot create projects if you have not created your devices or tablespaces.

NOTE For more information about space required for these devices, select Help | Search and enter the keywords *disk space*.



Sample Scenario

Having successfully logged in to the BBSD3 server, Brad got to work setting up projects. He had taken the precaution of generating a list of projects and users from his old Excel defect

tracking system. Once inside Tracker Administrator, he added his first project and started setting up the users for that project.

He had determined that the best way to work was to:

Pick a standard project,

Add all the custom user groups the administrators had agreed on in their meeting,

Then add all the users for that project.

Then he could create new projects using Tracker's project copy functionality (see ["Creating a New Project Based on an Existing One" on page 100](#)), and copy all applicable user groups and users over to the new project (see ["Copying Users" on page 119](#) and ["Copying User Groups" on page 152](#)).

Any changes to the user groups and users could be accomplished via the procedures ["Modifying User Group Permissions" on page 135](#) and ["Modifying Users" on page 116](#). (See page 107 for a continuation of this scenario.)

Creating a Project

In this procedure, all the work is done by the Server Administrator using Tracker Administrator—creating the project database, specifying the devices, and adding the project to Tracker.

- 1 Login to Tracker Administrator as Server Administrator. Use the *sa* or *system* DBMS login and password.
- 2 Click the Projects button.
- 3 Click the Add button.



- 4 Enter the project name. Enter the database name.

NOTE When entering a project name, refrain from using ampersands (&) or other special characters. Special characters are interpreted differently in the Tracker web client and will prevent users from seeing the project.

- 5 Select the Basic Project option from the Create group.

6 Choose OK.

If you are using ... Then ...

MS SQL Server,
MSDE, or Sybase You will see:

Enter data and log file names, and be sure to append the names to the **Filename** text boxes.

Oracle

You will see:

See step 8.

- 7 Select a device or tablespace for your project data and specify the amount of disk space you want to allocate. See page 293 for recommendations on space allocations.
- 8 Except on SQL Server, Select a device for your log data and specify the amount of disk space you want to allocate.

NOTE Oracle's temporary tablespace is not used by Tracker in this release. For more information, see ["About Devices, Tablespaces, and Datasources" on page 64.](#)

9 Click OK.

Upgrading Your Projects

Tracker 8 introduces issue types and issue relationships. With these features and the existing dependent and transition fields, you may want to create a plan to figure out the best way to use these features before upgrading your projects and rolling out Tracker 8.

When You Upgrade

When you upgrade to Tracker 8 you can:

- Define new issue types for new and existing records.
- Create issue relationships within and across projects.
- Clone records within a project.
- Copy records across projects.
- Use LDAP as a login source.
- Use single sign-on authentication.
- Use an upgrade to the Notify rules including the Escalation Rule enhancement.

NOTE This upgrade contains the ability to create dependent and transition fields. See [“About Dependent Field Relationships” on page 228](#) and [“Transition Fields” on page 213](#).

If You Don’t Upgrade

If you don’t want to upgrade your projects, you can continue to use your 7.x projects in Tracker 8.0, but you will not be able to use all of the new features. You can, however, use the features that don’t require a schema upgrade, which include:

- Printing, column sorting, manual notify, and attached files in the Tracker web client.
- Notify as a Windows service.

Discontinued Support for dBase in Tracker 8.0

The Tracker Learning Projects now uses MSDE. The development of new features is not designed for use with dBase. If you were using dBase for your production database (which is not recommended), you need to either:

- Copy your dBase project over to a Tracker 8.0 supported database to use the new features.
- Use the project from the Tracker 8.0 clients without use of the new features.

Before You Upgrade

To use the new features in Tracker, you must first plan out how you would like to use Tracker. You can define issues to be issue types and relate these types in defined relationships to create a

workflow. All of this work should be planned out so you can fully take advantage of the new features.

Refer to [“About Dependent Field Relationships” on page 228](#) and [“Transition Fields” on page 213](#) for more information on Tracker dependent and transition fields.

For more information on issue types and issue relationships, see [“About Issue Types” on page 186](#) and [“About Issue Relationships” on page 197](#).

Will you upgrade one project at a time or all of them at once?

You may want to upgrade to a separate database instance and use it as a test project to see how your implementation of the new features works in your environment. If you don’t upgrade all the projects at once, Tracker 7.x projects are accessible through Tracker 8.0 clients.

Example

You can use one project as a test project to explore the new features. When you get an understanding of what issue types and relationships are and how they can work with dependent and transition field values, you can begin to implement the features on the test project. Then you can see how the new features work in your environment and develop a plan to upgrade the rest of your projects.

How will you define issue types and relationships?

What is the best implementation for your environment? Review the documentation to understand how issue types and relationships work. Take a look at your projects to see what kinds of issues are reported and what issue types you can use for those issues. Define your process with relationships and cross-project relationships if necessary.

Example

If you use Tracker to report defects, enhancements, and tasks, you can create issue types for each of those types of records. You can define the workflow so that an enhancement transitions to a task and the initial records are related to one another in a parent-child relationship.

Will you create new Tracker projects with issue types and relationships or update the older projects with these new tools?

Creating a new project gives you the flexibility to define a workflow and create the issue types and relationships to support it. Working with existing projects requires combing through the records to see which records belong to what issue types. Whichever path you choose, you need to define the workflow process with relationships and issue types before you begin.

Example

If you have a project with a lot of history you may want to categorize the different records into issue types in order to maintain the history, and then relate these issue types together. If you are beginning a new project, you can create new issue types based on your work patterns from older projects.

With the ability to create issue types, does consolidating your Tracker projects into one project make sense?

If you have different Tracker projects for different types of issues, you can copy sets of records over to a new or existing Tracker project and define issue types for the kinds of issues that exist. See the *Merant Tracker User's Guide* to learn more about copying issues across projects. When you do this, the change history is not copied over.

Example

If you have one project for asset management, one for enhancements, and one for defects, you can copy these records

into an existing project and assign the copied records as issue types. You can also keep the projects separate, and relate the issue types across project boundaries instead of consolidating them.

How the Upgrade Affects the State Field

Tracker uses the State field to define issue type lifecycles. When you upgrade a project, you can let Tracker Administrator move an existing Transition field so that your user-defined choices, including transition rules and values, can be converted to the state field.

Converting the State field:

- Is necessary when the project already has a Workflow field defined.
- Copies all choices to the state field.
- Copies all transition and dependent field rules to the state field.
- Adds this change to the issue's change history.

When you select the option **Convert Existing Field to State Field**, Tracker Administrator will copy choices from the selected field to the State field during the upgrade. If the State field is currently a child field, any dependencies that listed the State field as the child will be deleted.

NOTE When you upgrade a project, the Public user group is given field permission for the State field.

How to Upgrade Your Projects

Before you upgrade your projects, you must first upgrade the TRKMASTER. The Tracker Master is the database where the server wide data is stored. You upgrade the Tracker Master by selecting the Tracker Master checkbox on the Upgrade dialog box.

To upgrade your project(s):

- 1 Deny logins and ensure that no users are currently logged in to the project(s) that you want to upgrade; then select Tools | Upgrade Projects.
- 2 In the Upgrade Projects dialog box, select the **Upgrade Tracker Master** check box to upgrade the master database. You must upgrade the database prior to upgrading a project. You can upgrade several projects at a time using this dialog box.
- 3 Select one or more projects to upgrade.
- 4 Select the **Convert Existing Field to State Field** check box if you have existing field choices that you want to convert to State fields for use with issue types. Selecting this option will open a new Upgrade Project <project name> dialog box for each project you are upgrading.

NOTE When you select this check box, the choices and transition rules from the selected field and its dependencies are copied to the State field and the State field's current value is modified to the value of the field that is being converted. Any values of <<None>> are converted to the Open choice.

- 5 Click OK. The Upgrade Project <project name> dialog box appears.

- 6 In the Upgrade Project dialog box, select the choice field that you want to copy to the State field during the upgrade.
- 7 Click OK. You are now ready to implement dependent and transition fields.

Refer to [“About Dependent Field Relationships” on page 228](#) and [“Transition Fields” on page 213](#) for more information on Tracker dependent and transition fields.

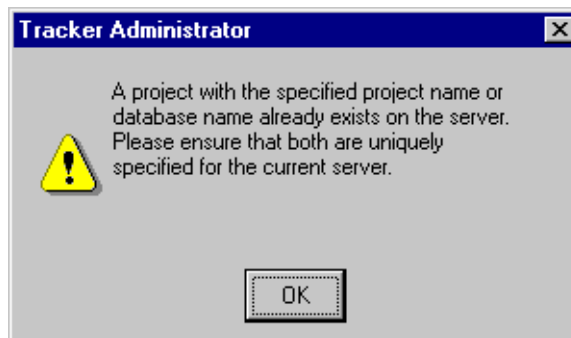
For more information on issue types and issue relationships, see [“About Issue Types” on page 186](#) and [“About Issue Relationships” on page 197](#).

Reattaching an Existing Database

This procedure adds an existing database to Tracker’s list of projects so that Tracker can recognize it as a valid project database. This feature is useful if you have restored from backup a project that was deleted from Tracker, or anytime that you have a valid Tracker database that Tracker does not recognize.

Follow [steps 1-4 on page 89](#); then:

- 1 Select Basic Project from the Create group.
- 2 Click OK. Tracker displays:



3 Click OK.

Tracker Administrator will now add the database to the list of Tracker projects without trying to create or initialize any tables.

Creating a New Project Based on an Existing One

This procedure copies everything in the project—users, user groups, permissions, and custom and dependent fields—into a new project. You can choose what to copy and what to leave out by selecting and deselecting check boxes. This procedure works only with Tracker version 3.0 projects or later.

Follow [steps 1-4 on page 89](#); then:

- 1 In the Create group, select Project Based On An Existing 3.x Project or Later Project.
- 2 In the Copy From group:

To copy a project on ...	Select ...
The Current Server	Current Server; then select the project from the Project list.
A different server	Other Server; then log in to the server and project you want to copy.
- 3 In the Copy Options group, select the parts of the project you want to copy. For example, if you want to copy user groups but not the users within them, select User Groups and clear Users.

NOTE Relationship records can only be copied if the project is being copied onto the same server or if the relationship record exists entirely within a project. You cannot copy a relationship across server boundaries even if both projects will ultimately exist on the target server.

NOTE If you do not copy groups or users, the resulting project will not contain any notification rules—not even the default rules.

- 4 Click OK to start the copy process.

NOTE If you don't copy issues from the other project, you need to reset Record ID for the new list of issues to start at 1 in sequence.

Allowing Special System Permissions for Project Administrators

When you copy existing relationship records into a new project, the Project Administrator can be granted expanded permission to be able to repair project databases and manage the system fields and relationships. Only a System Administrator can grant this permission.

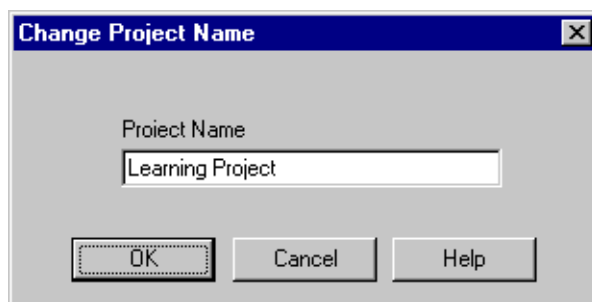
To expand permission for Project Administrators, choose Tools | System Permissions and select permissions in the System Permissions dialog box.

Renaming an Existing Project

Though you can have projects that are available only to either web client or desktop client interface users, all renaming must be done by an administrator in Merant Tracker Administrator.

To rename a project:

- 1 If you are renaming a project that has been enabled for web access,
 - a Log in to the web client through the web.
 - b Access the Web Client Admin Configuration Page.
 - c Disable the project you want to rename.
- 2 In the desktop client interface, click the Projects button.
- 3 Do one of the following:
 - Select the project you want to rename from the Current Projects list; then click the Modify button.
 - Double-click the project you want to rename from the Current Projects list.
- 4 Specify a different name for this project in the Project Name field.



- 5 Click OK.

- 6 If you are renaming a project that has been enabled for the web, log back in to the web client and enable the renamed project from the Web Client Admin Configuration Page.
- 7 Repeat steps 2 – 6 for each project you want to rename.



NOTE To refresh the list of projects to include projects that have been added or deleted since you opened a task window, click the Refresh button.

Deleting a Project

When a project is finished, you may want to delete the project database to free up disk space. (Alternately, you may want to keep the project online as a source of statistics for reporting a history of development efforts.)

NOTE Back up the project you are deleting before you begin. If the project is enabled for web client access, you must disable it before deleting it.

To delete a project:

- 1 Back up your existing Tracker database files.
- 2 Log in to Tracker Administrator as the Server Administrator.
- 3 Click the Projects button.
- 4 Select the project you want to delete.
- 5 Select Manage | Deny Logins to Selected Project.
- 6 Click the Who button. If Who indicates there are users logged in to this project, select Manage | Force Off. Wait for five minutes to allow Force Off to take effect.





7 Click the Delete button.

8 When you are prompted to confirm the deletion, click Yes.

NOTE You can switch to another project by selecting a different project name from the Project list.

9 Repeat steps 1 – 8 for each project you want to delete.

NOTE If you receive an ODBC error indicating that the deletion was not successful, restore the project you were trying to delete from your most recent backup and start again at step 2.

6 Creating and Managing Users

In this Chapter

For information about . . .	See page . . .
Overview	106
About Users	106
Adding Users	107
Setting Up Notify Options for a User	112
Setting Up Mail Options for a User	114
Setting Up Project Administrators	116
Modifying Users	116
Deleting Users	117
Recovering Deleted Users	118
Copying Users	119

Overview

Who Should Read This Chapter?	This chapter is intended for the Server Administrator <i>and</i> Project Administrators.
Why Read This Chapter?	This chapter contains instructions for creating and maintaining Tracker user IDs and user groups, and assigning user group permissions.

About Users

Who is a User?	A <i>user</i> is a Tracker user who is authorized to access a Tracker project. All Administrators (Server and Project) must be added as Tracker users. Usually, only a small number of the Tracker users on any server are Administrators. Usually, only one of the users is a Server Administrator.
Why Do We Have User IDs?	Each user must have a unique Tracker user ID. Tracker uses this ID to track system change requests. You can specify a password for each user if you want additional security. Users must enter these user IDs and passwords each time they start Tracker. Requiring users to log in protects project data and ensures that issues are associated with the users who submit, own, and update them.

NOTE If you are using MS SQL Server, Sybase, or Oracle, be aware that the Tracker user IDs are different from database (DBMS) user IDs. For more information see [“About the Tracker-DBMS Login Relationship” on page 66](#).

What Information is Associated with a User ID?	You can also choose to enter information about how to contact a user. A user ID is associated with the user’s telephone number, e-mail address, and other means of contact. This information includes an e-mail address that Notify uses to send automatic notifications via your corporate electronic mail system. If you’ve
--	---

What is a User Group?

already added users to a project, you can copy all or some of them to a new project. For instructions, see [“Copying Users” on page 119](#).

While adding a user, you also assign them to one or more user groups, such as Submitter, Development Engineer, or Administrator. Each user group is associated with a set of permissions that controls the actions of the members within the group. You can add a user to multiple groups so that they have a superset of permissions for all groups of which they are a member. For more information about user groups, see [“About User Groups and Permissions” on page 132](#).



Sample Scenario

Once all the projects and their related user groups and users were set up, Brad called each Project Administrator and asked him or her to log in and check his projects. He e-mailed each Project Administrator his project users' IDs and told him that all passwords were set to "password". Tracker seemed to be settling into its corporate environment very smoothly! (See [page 151](#) for a continuation of this scenario.)

Adding Users

This section provides procedures for adding users on all DBMS types. Be sure to use the table below as a guide in deciding which procedures to use for your specific setup and DBMS type.

Adding Users to the DBMS

In Tracker, users exist in two locations: on the DBMS and in Tracker. Some organizations may choose to have a DBMS login and password and a Tracker login and password for *each* Tracker user. You can create users first in Tracker or on the DBMS. They will not be able to log in until they are associated with both a Tracker login and a DBMS login.

How you add users is dependent upon your database type, the setup method you used, and whether you have *sa* privileges.

If your DBMS type is . . .	You used this setup option . . .	Use the Adding Users procedure on page 110 and . . .
MS SQL Server, MSDE, or Sybase	Default login	No other tasks.
	Multiple login	"To add users to a SQL Server or a Sybase database:" on page 109.
Oracle	Default login	"To add users to an Oracle database:" on page 110.
	Multiple login	Have your DBA associate each new user with the appropriate project database.

DBA User Creation Tasks

To add users to a SQL Server or a Sybase database:

- 1 Get a list of users associated with their particular project databases.
- 2 Use your DBMS tools to create and run the following script, which:
 - Adds all the new user logins to the DBMS.
 - Adds all the new logins to *TRKMASTER* and associates them with *PVCSTRKUSR*.
 - Adds specific users to the specific project databases they will be using.

```
use master
go

/* add users login to the server */
sp_addlogin user1_login, user1_password
go
sp_addlogin user2_login, user2_password
go
/* add users login to trkmaster database */
use trkmaster
go
sp_adduser user1_login,user1_login,pvcstrkusr
go
sp_adduser user2_login,user2_login,pvcstrkusr
go
/* add users to their project databases */
use trk_prj1
go
sp_adduser user1_login,user1_login,pvcstrkusr
go
use trk_prj2
go
sp_adduser user2_login,user_login,pvcstrkusr
go
```

- 3 To add the database server login to a new project database, create and run the following script:

```
use new trk_proj_1
go
sp_adduser user1_login,user1_login,pvcstrkusr
go
```

To add users to an Oracle database:

- 1 Using your Oracle tools, create user *Tracker1* identified by the password of your choice.
- 2 Grant connect to *PVCSTRKUSR* to *Tracker1*.
- 3 Repeat steps 1 and 2 for each user.

Adding Users in Tracker



To add users:

- 1 Click the Users button. The Users window appears.



- 2 Click the Add button. The User Options dialog box appears.

User Options - Software Learning Project

User Identification | Automatic Notification | Manual Notification

User Id:

Full Name:

Password:

Retype Password:

E-Mail:

Telephone:

User Groups:

- Administrator
- Development Engineer
- Documentation Engineer
- Product Manager
- Project Manager
- Public Queries/Reports**
- Quality Engineer
- Submitter
- Support Engineer
- Support Engineer Manager

All users belong to the Public User Group.

OK Cancel Help

- 3 Specify a user ID of 32 characters or less without spaces in the User ID field.
- 4 (Optional). Specify a password for this user in the Password field.
- 5 If you are changing the password, you must reenter the new password in the Retype Password field.

NOTE In most network environments, users must log in to the network before accessing network data. If you're using Tracker on a network, remember that Tracker user IDs are completely separate from network user IDs. Tracker does *not* detect network user IDs and use them as default user IDs.

- 6 (Optional). Specify other means of contacting this user in the Full Name, E-Mail, and Telephone fields.

NOTE You must enter the user's name in the Full Name field for it to appear on, for instance, issues and reports. The e-mail address must be the user's corporate mail system login ID in order for the user to receive either manual notification from Tracker or automatic notification from Notify.

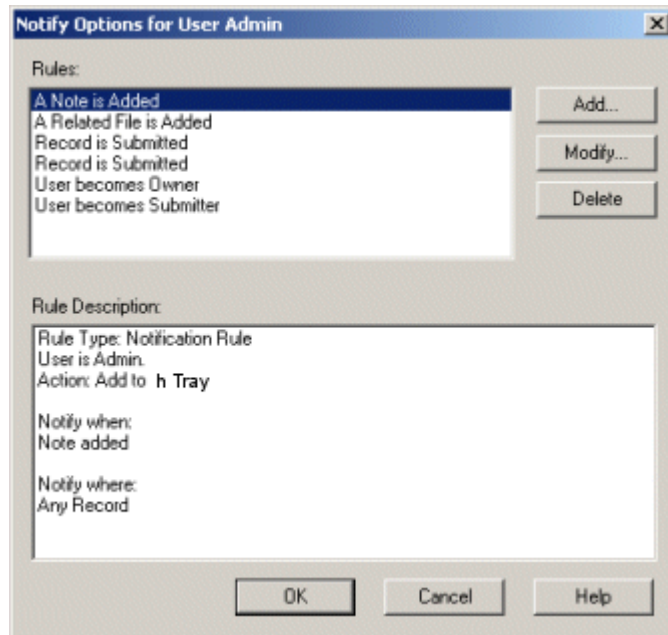
- 7 Select one or more user groups of which this user will be a member from the User Groups list.
- 8 Click Add.
- 9 To add Notify options, skip to ["Setting Up Notify Options for a User" on page 112](#). If you want to add Mail options, skip to ["Setting Up Mail Options for a User" on page 114](#).
- 10 Repeat steps 2 – 9 for each user you want to add.

Setting Up Notify Options for a User

You can send issues to a user's In Tray or, if you have Notify installed, as an e-mail message. After you have set up the user's options, you can modify those options at the user's request.

To set up Notify options for a user:

- 1 Click the Users button.
- 2 Select the name of the user you want to change in the Current Users list.
- 3 Select **Manage | Notify Options for User**. The Notify Options dialog box appears.



- 4 Click the Add button. This opens a pop-up menu with two options. Choose **Notification Rule** to open the Define Notify Rule dialog box.

NOTE You can also choose **Escalation Rule** to set up a rule using time-based and/or negative conditions (for example, seven days pass with no change to an issue). If the user doesn't have Escalation Rule permission, the Add button directly opens the Define Notify Rule dialog box.

- 5 In the Title text box, enter the title of the notification rule. This appears in the subject line of an automatic e-mail notification wherever the \$RULE macro occurs.
- 6 Optionally, click the When button to customize the change condition displayed in the Notify When box.
- 7 Click the Where button to customize the query on which the rule is based, or click **Insert** to choose an existing query from a list.
- 8 Select the Action tab.
- 9 Select the In Tray option if you want to either send notifications via the user's In Tray, or remove the record from the user's In Tray. Specify **Add** or **Remove**.
- 10 Select the E-Mail option if you want to send notifications via e-mail. This option requires Notify.
- 11 Select the **Do not notify if user is author** option if you want to suppress notification of changes the user makes. This prevents these notifications from taking up space in the user's In Tray or e-mail.
- 12 Click OK.
- 13 Repeat steps 4–12 for each notification you want to set. The notification options will appear in the Notify Options dialog box.

Setting Up Mail Options for a User

After you set up a user, you can set up the user's mail options. Use the following procedure to designate the style sheet, message subject, and other message options used in manual and automatic e-mail notifications. Automatic e-mail notification requires the Notify application.

To set up mail options for a user:

- 1 Click the Users button. The Users window appears.
- 2 Select the name of the user you want to change in the Current Users list; then click the Modify User button. The User Options dialog box appears.
- 3 Select either the Manual Notification tab or the Automatic Notification tab.
- 4 In the Style Sheet group, select a style sheet to use when the message is formatted for the mail system.
- 5 In the Subject for... group, enter a string to use as a template for the mail message subject when:
 - A single issue is sent via e-mail.
 - Multiple issue are sent via e-mail.

NOTE You can use control codes (prefixed with a \$) to refer to specific information about the current project or issue.

- 6 If you selected Manual Notification, you can include a CC: list by clicking Modify in the CC: group and then selecting users to add to the notification list.
- 7 Again in Manual Notification only, you can select options to send a message as urgent, to request a return receipt, and so on. For more information, select Help | Search, and enter the keyword *mail*.
- 8 Click OK.
- 9 Repeat steps 2 – 8 for each user you want to set mail options for.

Setting Up Project Administrators

The Server Administrator can designate Project Administrators for specific projects by assigning certain users to the Administrator user group. The Project Administrators can then perform *most* of the Administrator tasks on those projects.

Before you can log in as a Project Administrator, the Server Administrator must have already done the following:

- Added you as a user and optionally assigned you a Tracker password. See “Adding Users” on page 107.
- Assigned your user ID to the Administrator Group for each project you will administer. See “About User Groups and Permissions” on page 132.

—or—

Copied your user ID to the projects for which you are allowed to perform administrative tasks. See “Copying Users” on page 119.

Modifying Users

Use this procedure to change information about a user’s login ID, name, e-mail address, or telephone number. You can also change a user’s password if they have forgotten it or add them to a different user group. This can be done while the user is logged in.

To modify information about a user:



- 1 Click the Users button. The Users window appears.
- 2 Do one of the following:



- Select the name of the user you want to change in the Current Users list; then click the Modify button.
- Double-click the name of the user you want to change in the Current Users list.

In either case, the User Options dialog box appears.

- 3 Use the User Options dialog box to change information about this user.
- 4 Click OK.
- 5 Repeat steps 2 – 4 for each user you want to change.

Deleting Users

Use this procedure to delete a user ID for a user who has left the project. This can be done while the user is logged in.

When you delete a user ID, no one can log in with that user ID; however, the information about the user remains in the database, so you can recover it if the user returns to the project. For information on recovering users, see [“Recovering Deleted Users” on page 118](#).

NOTE Deleted users will still show up in change history.

To delete a user:



- 1 Click the Users button.
- 2 Select the user ID you want to delete.



- 3 Click the Delete button.
- 4 When you are prompted to confirm the deletion, click Yes.
- 5 Repeat steps 2 – 4 for each user you want to delete.

Recovering Deleted Users

When a user is deleted from a project, the system actually flags them as inaccessible. They are not deleted from the database. Use this procedure to recover a user ID if a user is returning to a project, or if you accidentally deleted the user.

NOTE If you recover a user who has returned to the project, we recommend that you review the user ID information for accuracy.

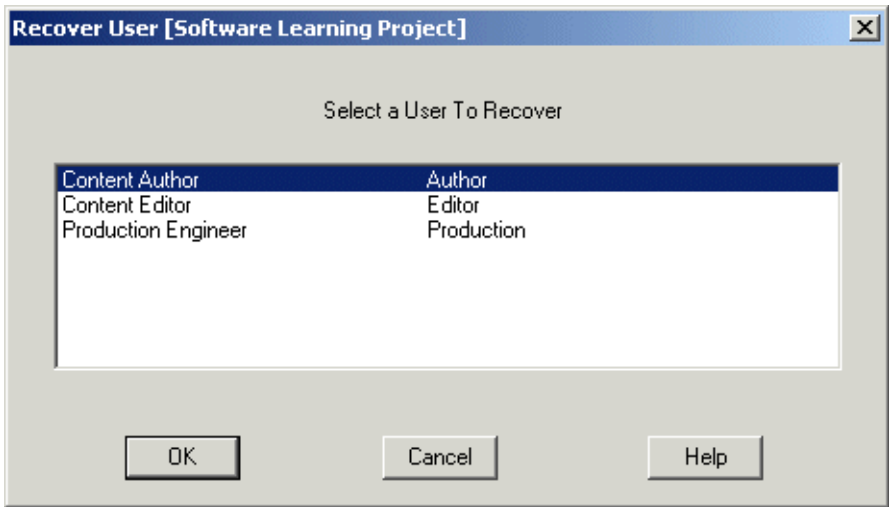
To recover a deleted user:



1 Click the Users button. The Users window appears.



2 Click the Recover button.



3 Select the name of the user you want to recover; then click OK.

4 Repeat steps 2 – 3 for each user you want to recover.

Copying Users

Use this procedure to copy users to other projects so you won't have to add them again. This saves time when you set up users for multiple projects.

To copy a user:

- 1 Log in to all the projects you intend to copy users to and from.

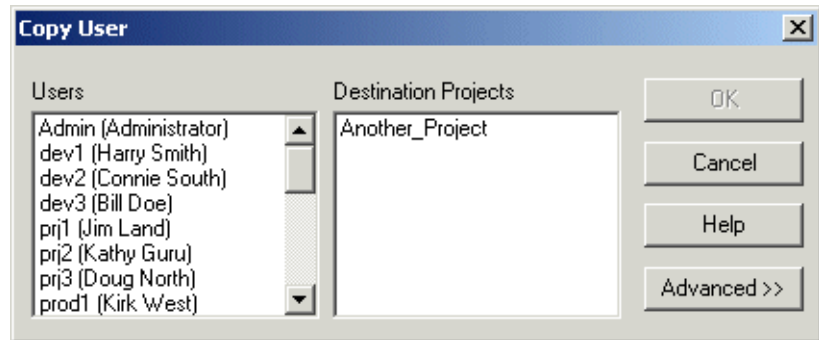
NOTE We recommend that you force off the users in the projects you are copying to and from. See [“Controlling Logins”](#) on page 286.



- 2 Click the Users button.
- 3 Select the project you want to copy from the Project list in the toolbar.



- 4 Click the Copy button.



- 5 Select the users you want to copy in the Users list.
- 6 Select the projects to which you want to copy these users in the Destination Projects list.

- 7 Choose Advanced >> to specify additional options for copying users. For more information, select Help | Search and enter the keyword *Copy User*.
- 8 Click OK.

Enabling Single Sign-On for Users

Single sign-on is only active when a user has been granted a license on his or her machine. When a user changes to a different application with single sign-on enabled, the single sign-on username is used to log in to the application instead of the most recently used login name for the project.

As long as a license is active, the user does not need to enter a username and password again.

To enable single sign-on:

- 1 From the Project window, choose Manage | Configure Authentication. The Configure Authentication dialog box appears.
- 2 Select the Enable single sign-on check box to configure single sign-on for users of all projects in the Tracker server.

NOTE All Tracker and Version Manager interfaces can share the single sign-on information set in this dialog box and in the Version Manager Configure Project Database dialog box, General tab. Enable single sign-on in all products to allow users to switch between applications without having to log in again.

For more information about configuring authentication for users, see [“Configuring Authentication” on page 123](#).

7 Setting up LDAP for Login Authentication

In this Chapter

For information about . . .	See page . . .
Overview	122
Using LDAP in Tracker	122
Configuring LDAP in Tracker	122
Setting Up Redundant and Distributed Servers	129

Overview

Who Should Read This Chapter?	This chapter is intended for the Server Administrator.
Why Read This Chapter?	This chapter discusses setting up and using Lightweight Directory Access Protocol (LDAP) in Tracker for user authentication, single sign-on, and redundant and distributed servers.

Using LDAP in Tracker

Tracker uses one of two login authentication sources: Tracker user ID or LDAP server. By default, Tracker authenticates users against the Tracker user ID. This chapter discusses how to set up and use [LDAP](#) with Tracker.

LDAP is an industry-wide standard, and many online directories, such as Active Directory, conform to it. LDAP provides login information from compliant sources and allows you to share user authentication between Tracker and Version Manager.

Configuring LDAP in Tracker

As a Tracker administrator, you configure how LDAP is used in Tracker through the Manage menu from the Project window. If the current project is earlier than Tracker 8.0, LDAP configuration is not available.

Configuring Authentication

The settings in the Configure Authentication dialog box apply to all the projects on the selected Tracker server. The server does not allow some projects to authenticate against Tracker user ID and other projects to authenticate against LDAP.

To configure authentication:

- 1 From the Project window, choose Manage | Configure Authentication. The Configure Authentication dialog box appears.

Configure Authentication

☒ Enable single sign-on

OK

Cancel

Help

Authenticate against

☐ Tracker user list

☒ LDAP server

LDAP Configuration

☐ Use SSL (Secure Socket Layer)

☐ Follow referrals

Base user context:

User naming attribute:

☒ Bind anonymously

User DN:

Password:

Connection timeout:

30

 seconds

LDAP attributes to use for new Tracker user (optional)

Full name:

E-mail address:

Telephone number:

NOTE Single sign-on is available whether you use LDAP server or Tracker user list authentication. For information about enabling single sign-on, see [“Enabling Single Sign-On for Users” on page 120.](#)

- 2 Choose whether to authenticate against the Tracker user list or the LDAP server.
 - 3 Select the Use SSL check box to enable the Secure Socket Layer.
-

NOTE Referrals are used for distributed LDAP server optimization. For more information about the Follow referrals option, see [“To configure distributed LDAP servers in Tracker:” on page 130.](#)

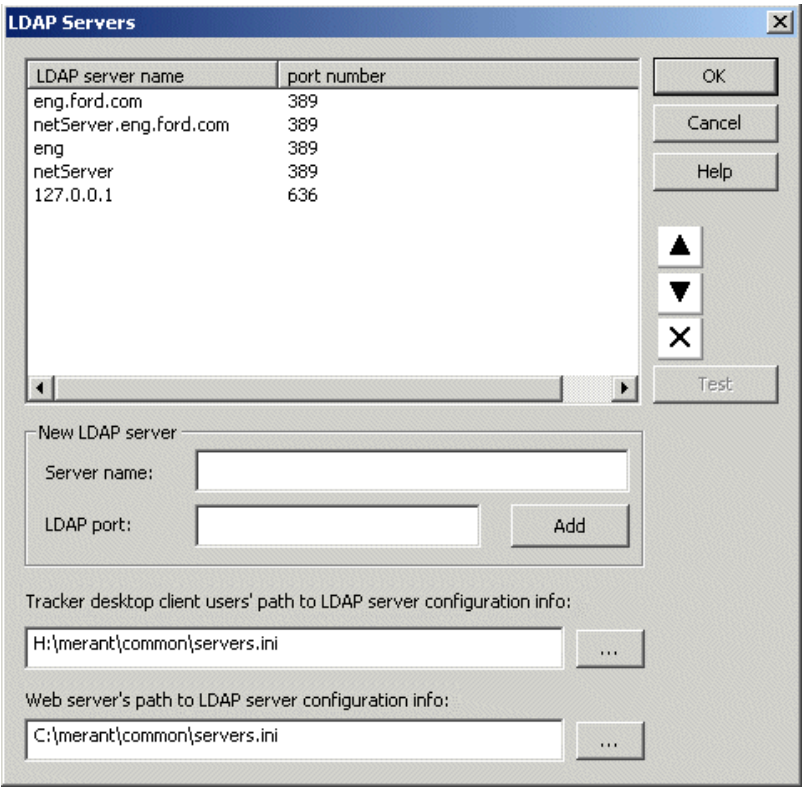
- 4 Specify the base user context (distinguished name) in the Base User Context field. This is the base from which to search for users.
- 5 Select the user naming attribute in the User Naming Attribute field. The LDAP server holds the user ID value in this attribute.
- 6 Select the check box to bind anonymously, or enter a user DN and password.
- 7 Set the connection time limit.
- 8 Optionally, choose LDAP attributes to use for new Tracker users. These settings allow you to select the names of LDAP server attributes from which to get a Tracker user’s full name, e-mail address, and telephone number when creating a new user.

Configuring LDAP Servers

The LDAP server information you set in the LDAP Servers dialog box applies to all the projects on the Tracker server where you are currently logged in.

To configure LDAP servers:

- 1 From the Project window, choose Manage | LDAP Servers. The LDAP Servers dialog box appears.



- 2 View the list of servers and port numbers. Tracker attempts to bind to the LDAP servers to authorize login in the order in which the servers appear in the list.



- 3 Click the up arrow icon or the down arrow icon to move a selected server to another position in the list, or click the X icon to delete a server from the list.
- 4 Enter a name and port number in the New LDAP server section and click the Add button to add a new server.
- 5 Enter a path to LDAP server configuration information for Tracker users. The same .INI file may be used for Tracker and Version Manager.

NOTE If you have both desktop and web client users, you can use the same servers.ini file on the web server machine for both. See recommendation for setting up paths to servers.ini file for all users below.

- 6 Check the configuration of each LDAP server.
 - a Select the server you want to check.
 - b Click the Test button. Tracker will try to connect to the specified LDAP server and then report the results.
- 7 Click OK to finish configuration. If you click OK without specifying a path to LDAP server configuration information, LDAP information will be saved to servers.ini in the same Tracker directory as connect.ini.

To use the same servers.ini file for all Tracker users:

- 1 Store the LDAP configuration file on a hard drive attached to the web server machine (for example, C:\Merant\Tracker\servers.ini).
- 2 Set the web server's path to LDAP server configuration info in the LDAP Servers dialog box to the hard drive location (C:\Merant\Tracker\servers.ini).
- 3 Direct Tracker web client users to map a particular drive letter to the web server's hard drive (for example, H:).

- 4 Set the Tracker desktop client users' LDAP configuration file path to the same path on the server (H:\Merant\Tracker\servers.ini).

Configuring LDAP Users

Tracker User Groups control each individual user's privileges within Tracker. In the Configure LDAP Users dialog box, if the **Automatically create new users from LDAP as Tracker users** check box is selected, Tracker automatically creates a Tracker user with the same username as the user's LDAP login name the first time an authenticated LDAP user opens a Tracker project.

Because Tracker user permissions are assigned to groups, every new Tracker user is created in the group Public, which exists for all Tracker projects. You can add a new Tracker user to an additional group using the Configure LDAP Users dialog box.

- 1 From the Project window, choose Manage | Configure LDAP Users. The Configure LDAP Users dialog box appears.



- 2 Select the Automatically create new users from LDAP as Tracker users check box to create a Tracker user with the same username as the user's LDAP login name when the

authenticated LDAP user opens a Tracker project for the first time.

- 3 Optionally, you can select an additional user group for the new Tracker user and then click OK.

Setting Up Redundant and Distributed Servers

If you are using LDAP for authentication and your LDAP server is down, users cannot log in to Tracker. One solution is to set up more than one LDAP server, each containing all the same information or at least enough information to allow logins. These multiple LDAP servers with the same information are called [redundant servers](#). This configuration is common in Microsoft Active Directory installations.

Alternatively, you may have more than one LDAP server that you do not use redundantly. If your LDAP servers do not have all the same information on them, such as having a different sets of logins on each, they are called [distributed servers](#).

If you have redundant servers, you can set up LDAP configuration to search through an ordered list of servers if one is unavailable.

If you have distributed servers, you can set up LDAP to look for a login at more than one LDAP server and authenticate the login against whichever LDAP server stores the data for that particular user.

To configure redundant LDAP servers in Tracker to optimize login speed:

- 1 Follow the steps to configure LDAP servers on page [126](#).

- 2 Pay special attention to the ordered list of servers (step 2) and reorder them as needed.

To configure distributed LDAP servers in Tracker:

- 1 In the Configure Authentication dialog box, select the Follow referrals check box. See [“To configure authentication:” on page 124](#).

8 Managing User Groups and Permissions

In this Chapter

For information about . . .	See page . . .
Overview	132
About User Groups and Permissions	132
Modifying User Group Permissions	135
Creating User Groups	149
Deleting User Groups	151
Copying User Groups	152

Overview

Who Should Read This Chapter?	This chapter is intended for the Server Administrator and Project Administrators.
Why Read This Chapter?	This chapter explores how to create and maintain Tracker user groups and assign user group permissions.

About User Groups and Permissions

When you add a user to Tracker, you assign the user to one or more *user groups*. Members of a user group typically have the same job description and similar responsibilities. Each user group has a set of *permissions* that control what actions group members can perform. You can add a user to multiple user groups to provide them with a combination of permissions that is not limited to the set available to any one user group.

The user group determines which permissions this user will have to submit, own, and update issues, as well as update and run reports and queries. All users belong to the Public group by default. If you want a user to have more permissions than any one group has, you can add them to multiple groups.

NOTE The user ID permissions you create using Tracker Administrator are separate from the user ID permissions for the database that contains Tracker data.

After you add users, you can modify the permissions for any user group except Administrator. This allows you to control which Tracker functions group members can perform on issues. You'll also find instructions for defining custom user groups if the standard groups don't meet your needs.

NOTE To enable or disable access to a user group for an existing user, see [“Modifying Users” on page 116](#).

Standard User Groups

The standard user groups provided by Tracker already have a set of permissions defined for them. If the standard groups don't meet your needs, you can modify or create as many user groups as you wish using Tracker's extensive selection of permissions.

Tracker Administrator provides the following standard user groups:

This user group...	Allows...
Administrator	Server and Project Administrators all permissions for every project they are assigned to. It is not possible to change the permissions of the Administrator group. See NOTE below.
Development engineers	users to design, implement, or maintain software
Quality engineers	users to test and verify software against published standards
Public	standard access to which all users belong by default
Public Queries/Reports	users standard access to run public queries and reports
Submitters	users to test software and submit system change requests
Support engineers	users to provide product support to end users
Managers	users to oversee product development, quality assurance, and product support.

NOTE Server Administrator and Project Administrator both use the Administrator user group. However, you have to be logged in as Server Administrator to perform certain functions that cannot be done under a Project Administrator login.

Selecting Appropriate User Group Permissions

When assigning permissions, consider the job function that should be able to change it. For best results, refer to the implementation plan you created in [“Planning a Successful Tracker Implementation” on page 44](#).

The lifecycle of an issue typically includes these phases:

- The issue is submitted. A submitter enters the issue and may attach files or notes to help explain the change request.
- An owner is assigned. The issue is reviewed and assigned to a development or maintenance engineer for resolution. In some organizations, development engineers assign themselves issues that are relevant to their code as change requests appear in their In Tray.
- The issue is updated with new information. Data associated with the issue is added as further information becomes available about the change request and its resolution.
- The issue is resolved. When an issue is resolved and the resolution is verified, the record is closed. Some issues may be reopened if testing reveals that the resolution was not completely correct.

Available Permissions

The permissions available in Tracker Administrator are divided into these categories:

This permission type...	Controls a user's ability to...
Record	submit/update issues
Field	modify standard or custom fields in issues
Query	modify and run queries
Report	modify and run reports
General	edit project style sheets, build reports, or modify notify options.

NOTE For detailed information on specific permissions within these categories, select Help | Search and enter *permission* in the first field. Click OK for a list of permission-related topics.

Modifying User Group Permissions

Use this procedure to modify user group permissions when, for example, the user group's responsibilities change. If you have created a custom user group to run a specific report and that report has been split into three separate reports, you'll need to give the group permission to run each of the three new reports.

NOTE To enable or disable access to a user group for an existing user, see ["Modifying Users" on page 116](#).

Selecting a User Group to Modify

To select the user group you want to modify:



- 1 Click the User Groups button.
- 2 Do one of the following:



- Select the user group whose permissions you want to modify from the Current User Groups list and click the Modify User Group button.
- Double-click the user group whose permissions you want to modify in the Current User Groups list.

- 3 Use any of the following procedures to set permissions:

Setting General Permissions	137
Setting Record Permissions	139
Setting Field Permissions	143
Setting Query Permissions	145
Setting Report Permissions	147

NOTE For any of the procedures that follow, you can switch from one project to another by selecting a different project name from the Project list.

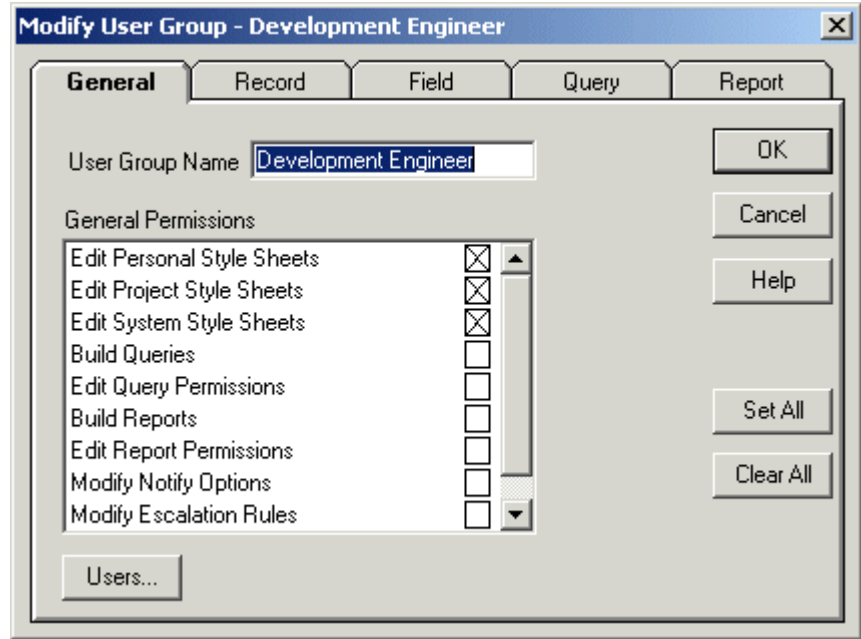
Project:

Software Learning Project ▾

Setting General Permissions

To set general permissions:

- 1 Select the General tab.



NOTE If you click the Users button, a dialog box that allows you to add users to, and remove users from, a user group, is displayed.

- 2 Select (or deselect) permissions.
 - To assign all permissions, click Set All.
 - To remove all permissions, click Clear All.
- 3 Click OK or select another permission category (tab).

Here is an explanation of each of the general permissions:

This permission . . .	Allows users to . . .
Edit Personal Style Sheets	Edit style sheets accessible to other users of this workstation.
Edit Project Style Sheets	Edit style sheets accessible to all users of this project.
Edit System Style Sheets	Edit style sheets accessible to all users in all Tracker projects.
Build Queries	Define queries to search for records that match certain criteria.
Edit Query Permissions	Modify permissions associated with queries.
Build Reports	Define Tracker reports.
Edit Report Permissions	Modify permissions associated with reports.
Modify Notify Options	Change their own e-mail and In Tray notification options.
Modify Notify Options	Modify options for notifying other users through Notify.
Modify Escalation Rules	Modify the escalation rules.
Change Issue Type	Change the issue type for any issue.
Add new relationship	Create a new relationship of any type.
Delete existing relationship	Remove existing relationships of any type.

Setting Record Permissions

Use record permission settings to control user access to different areas of Tracker. Most users will need the basic permissions. The more experienced or users with a more important role in the project will require more permissions.

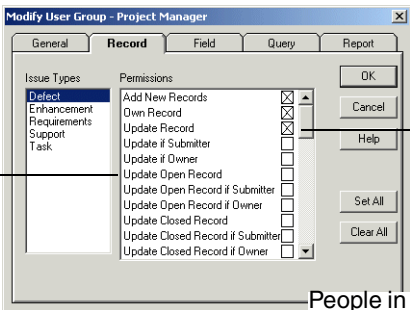
Record Permission Hierarchy

When changing permissions for updating records, it is important to realize that some options override others.

This permission . . .	Can be overridden by . . .
Update Record	cannot be overridden
Update if Submitter	Update Record
Update if Owner	Update Record
Update Open Record	Update Record, Update if Submitter, Update if Owner
Update Open Record if Submitter	Update Record, Update if Submitter, and Update Open Record
Update Open Record if Owner	Update Record, Update if Owner, and Update Open Record
Update Closed Record	Update Record, Update if Submitter, Update if Owner
Update Closed Record if Submitter	Update Record, Update if Submitter, Update Closed Record
Update Closed Record if Owner	Update Record, Update if Owner, Update Closed Record

*For example,
This scenario overrides permissions:*

If you deselect
Update Open
Record for a
given user
group ...

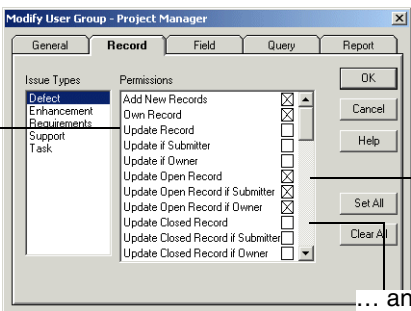


... but you leave
Update Record
selected ...

People in this user group can still
update open records because
Update Record takes a higher
precedence.

This scenario does not:

But by
deselecting
Update
Record ...



... selecting
Update
Open
Record ...

... and deselecting
Update Closed Record ...

... you make it possible for
users to update open records
only (not closed ones).

Permissions for Dependent Fields

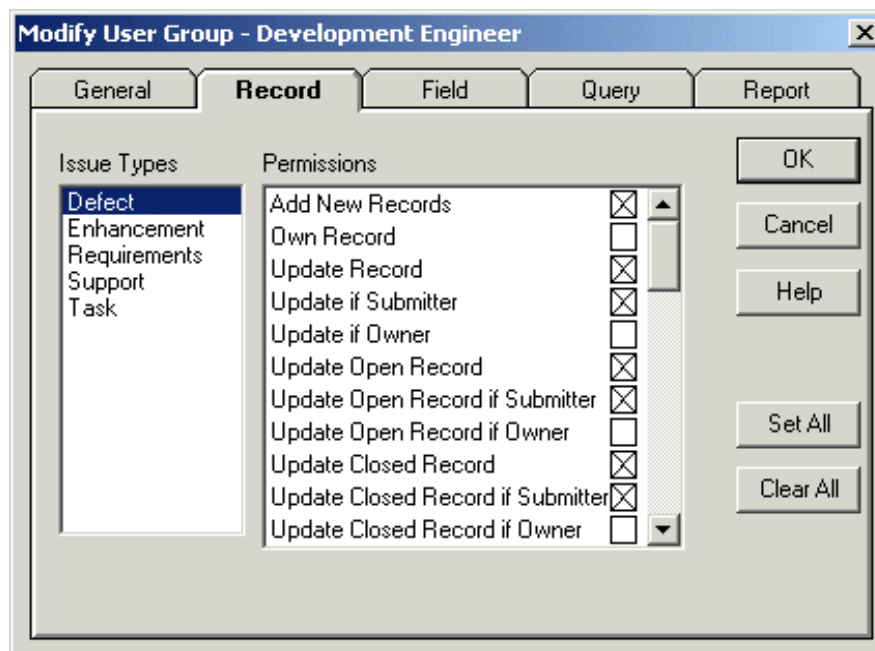
When assigning users the record permission “Override field mappings”, you are allowing the users to override the dependent and transition field relationships established for all dependent, transitional, and state fields. This will create invalid dependent relationships. This tool is useful in special circumstances when a project manager or administrator needs to update several issues at once for another user who is unavailable. This permission is only available in the Tracker desktop client from the Update Field dialog box.

When this permission is assigned, a check box is enabled on the Update Field dialog box in Tracker.

Refer to [“About Dependent Field Relationships” on page 228](#) for more information on implementing dependent fields.

To set record permissions:

- 1 Select the Record tab.



- 2 Select (or deselect) permissions.
 - To assign all permissions for this record type, click Set All.
 - To remove all permissions for this record type, click Clear All.
- 3 Click OK or select another permission category (tab).

NOTE Field permission settings will always override record permission settings. Also, some Update Record options are overridden by others. See ["Record Permission Hierarchy" on page 139](#) for more information.

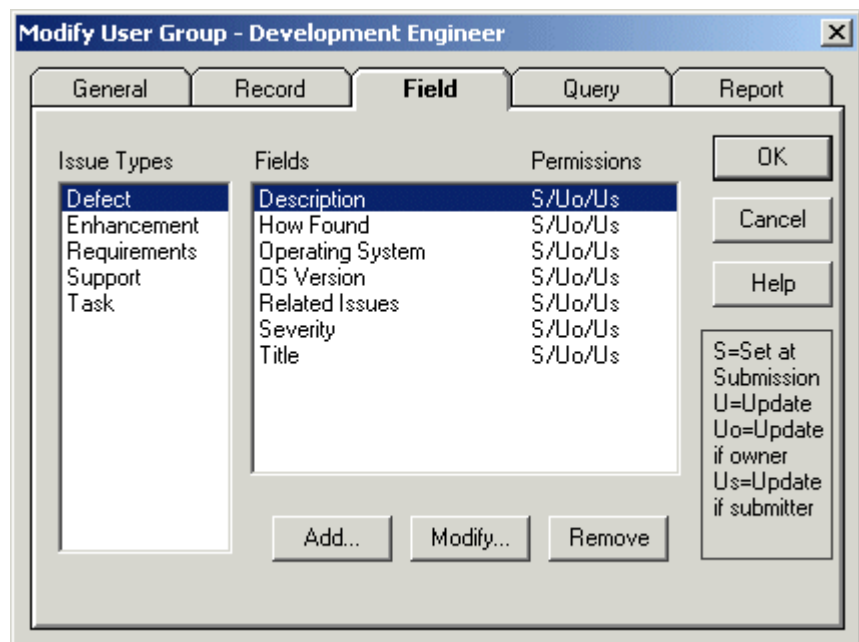
Setting Field Permissions

Field permissions that are set for a user group will override any record permissions that are set for the same group. Setting field permissions allows you to control user privileges on a field-by-field basis.

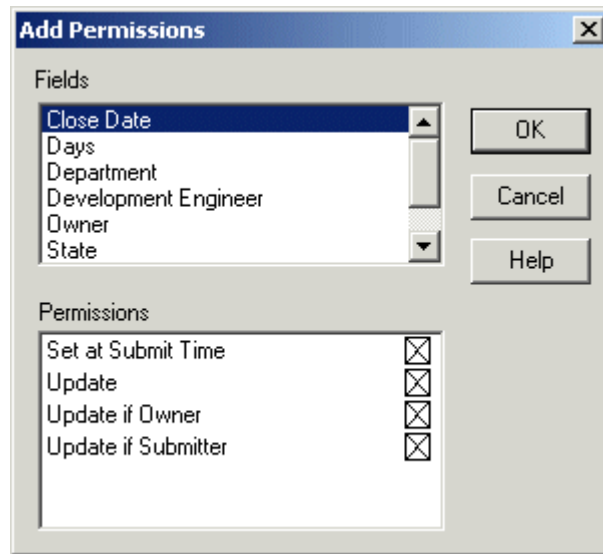
NOTE If a user does not have permission to see the field, it will not appear on their form, even if it is required.

To set field permissions:

- 1 Select the Field tab.



- 2 To set permissions for a field, do the following:
 - a Click Add.

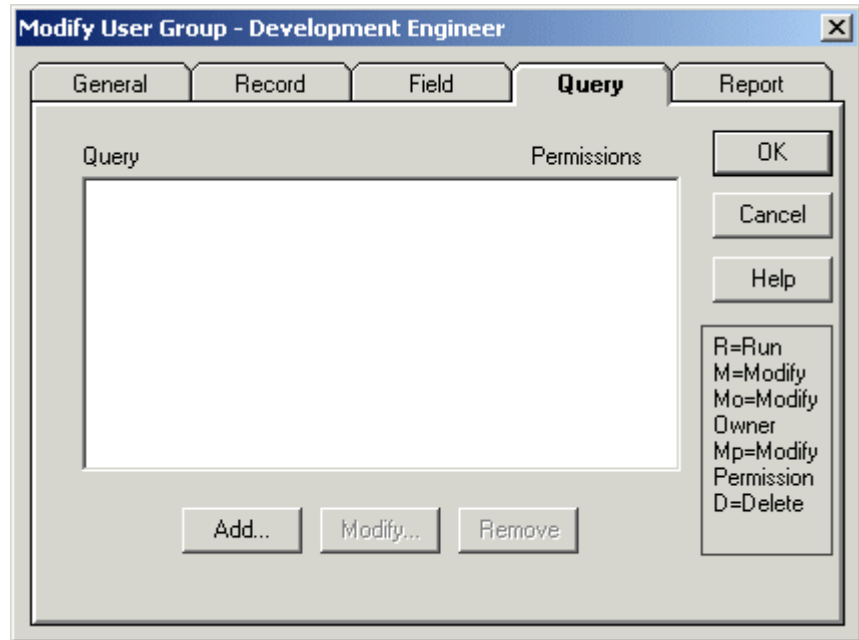


- b Select a field in the **Fields** list; then select or deselect permissions for the field in the **Permissions** list.
 - c Click OK.
 - d Repeat steps a – c for each field you want to set permissions for.
- 3 Select the field in the **Fields** list and click Modify to change the field's permissions, or click Remove to remove them.
- 4 Click OK or select another permission category (tab).

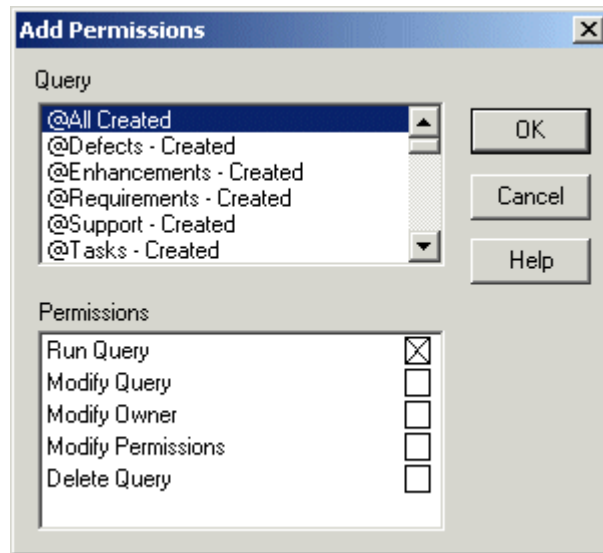
Setting Query Permissions

To set query permissions:

- 1 Select the Query tab.



- 2 To set permissions for a query, do the following:
 - a Click Add.

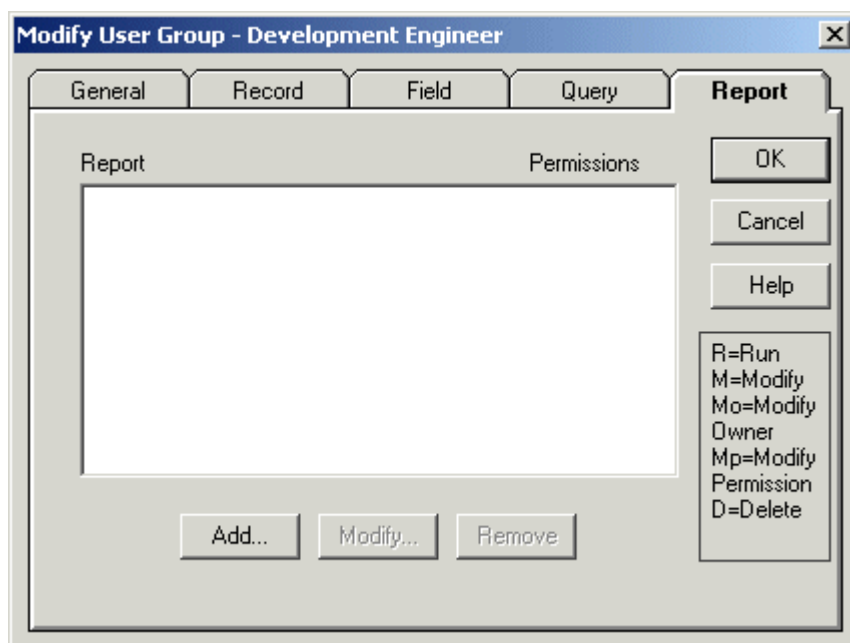


- b** Select a query in the Query list; then select or deselect permissions for the query in the Permissions list.
 - c** Click OK.
 - d** Repeat steps **a** – **c** for each query that you want to set permissions for.
- 3** To modify a query's permissions, select the query in the Query list and click Modify. Select or deselect permissions in the Modify Permissions dialog box; then click OK.
- 4** To remove a query's permissions, select the query in the Query list and click Remove.
- 5** Click OK to finish choosing permissions or select another tab to go to another permission category.

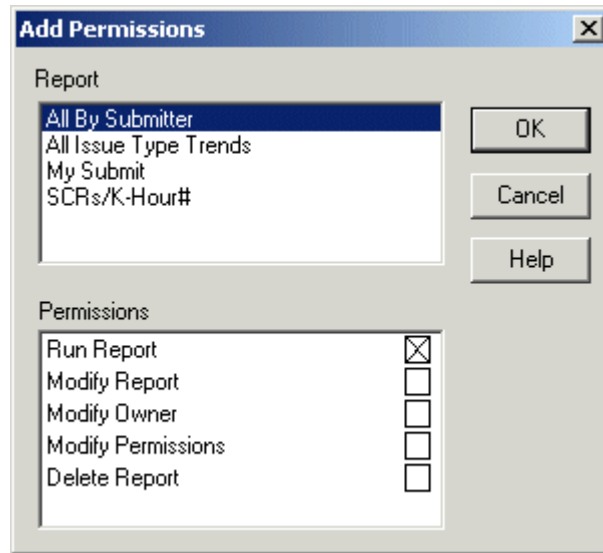
Setting Report Permissions

To set report permissions:

- 1 Select the Report tab.



- 2 To set permissions for reports, do the following:
 - a Click Add.



- b Select a report in the Report list; then select or deselect permissions for the report in the Permissions list.
 - c Click OK.
 - d Repeat steps a – c for each report that you want to set permissions for.
- 3 To modify a report's permissions, select the report in the Report list and click Modify. Select or deselect permissions in the Modify Permissions dialog box; then click OK.
- 4 To remove a report's permissions, select the report in the Report list and click Remove.
- 5 Click OK to finish selecting permissions or select another tab to go to another permission category.

Creating User Groups

Use this procedure to add a custom user group when, for example, a new job classification has been added to the QA function. If a new job has been created specifically to accumulate and interpret statistical QA information, you may want to create a custom user group that has permissions to run the appropriate reports.

To create a custom user group:



1 Click the User Groups button.



2 Click the Add button.

3 Enter the name of the new user group in the User Group Name field.

Add User Group

General | Record | Field | Query | Report

User Group Name:

General Permissions:

Edit Personal Style Sheets	<input type="checkbox"/>
Edit Project Style Sheets	<input type="checkbox"/>
Edit System Style Sheets	<input type="checkbox"/>
Build Queries	<input type="checkbox"/>
Edit Query Permissions	<input type="checkbox"/>
Build Reports	<input type="checkbox"/>
Edit Report Permissions	<input type="checkbox"/>
Modify Notify Options	<input type="checkbox"/>
Modify Escalation Rules	<input type="checkbox"/>

Buttons: OK, Cancel, Help, Set All, Clear All, Users...

NOTE If you click the Users button, Tracker displays a dialog box that allows you to add users to and remove users from a user group.

- 4 Use the Add User Group dialog box to specify the permissions for this group. See [“Modifying User Group Permissions” on page 135](#).
- 5 When you finish modifying permissions, click OK.
- 6 Repeat [steps 2- 4](#) for each custom user group you want to add.

Creating Token User Groups

When you use user fields as dependent fields, you can organize users from larger functional groups into smaller collections appropriate for a project. This will help you expand your user field values when using dependent fields.

These *token groups* can contain cross-functional groups but setting up these groups does not change the larger group that governs users' permissions.

Token Groups Example

A Tracker project has 3 user groups: Dev, Doc, and QA. Every Tracker user is assigned to one of these 3 groups and each group has a different set of permissions.

The Administrator has made the *Owner* field dependent upon the *Functional Area*. People who work on the UI, can only own issues in the UI functional area, but people in the Dev group are not the only users who “own” issues. After a developer fixes it, the QA person owns it to test it and the Doc person owns it to make sure it's in the on-line help.

The Administrator makes a token group, "UI", that has a Dev, Doc, and QA users. This is the group mapped to the Functional Area "UI". However, the UI group has no permissions associated with it - the users still get their permissions from their other core groups Dev, Doc & QA.

Deleting User Groups

Use this procedure to delete a custom user group when that set of permissions is no longer required by your organization and you are sure it will not be required again. For example, if you have created a custom user group to run a specific report, and that report has been identified as unnecessary, it may be time to delete the group.

To delete a user group:



- 1 Click the User Groups button.
- 2 Select the user group you want to delete from the Current User Groups list.



- 3 Click the Delete button.
- 4 When you are prompted to confirm the deletion, click Yes.
- 5 Repeat steps 2 – 4 for each user group you want to delete.



Sample Scenario

Later that day, Brad's boss called to tell him one of the Project Administrators had transferred to another division. All of his tasks needed to be reassigned to another team member who was not currently an Administrator—effective immediately.

Brad displayed both users in Tracker Administrator and copied the permissions from the reassigned Administrator to the new Administrator. Then he e-mailed the new Administrator to welcome him to the Administrator team and notify him of the change in his permissions. (See page 196 for a continuation of the scenario.)

Copying User Groups

To save time, you can copy user groups from one project to another.

To copy a user group:

- 1 Log in to all the projects you intend to copy user groups to and from.

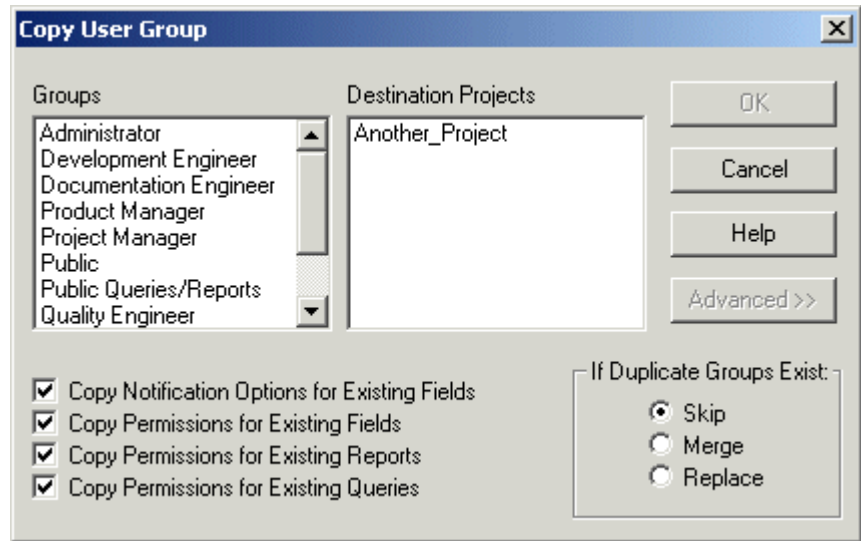
NOTE It is recommended, though not required, that you force off the users in the projects you are copying to and from. See [“Controlling Logins” on page 286](#).



- 2 Click the User Groups button.
- 3 Select the project you want to copy from.



- 4 Click the Copy button.



- 5 Select the user groups you want to copy in the **Groups** list.
- 6 Select the projects to which you want to copy these groups in the Destination Projects list.
- 7 Click Advanced >> to select other options for copying user groups. For more information, select Help | Search, enter the keywords *copying user groups*, and then select **Copy User Group Dialog Box**.

NOTE If you do not select advanced options, the related permission and/or rules are not copied to the new user group.

- 8 Click OK.

9 Preparing Data for External Reporting Tools

In this Chapter

For information about . . .	See page . . .
Overview	156
What the Assistant Does for You	156
Required Skills and Knowledge	157
Using the Assistant Under MS SQL Server	160
Using the Assistant Under Sybase	166
Using the Assistant Under Oracle	173
Creating Project Lists in the Report Assistant	179
Updating Views or Tables	180
Descriptions of the Views and Tables	180

Overview

Who Should Read This Chapter?	This chapter is intended for the Server Administrator.
Why Read This Chapter?	This chapter contains a description of the Report Assistant, a utility that prepares tables and views of project data for querying and reporting via third party tools such as Crystal Reports and Microsoft Access.

What the Assistant Does for You

The Assistant creates views and tables of project data.	The Report Assistant, located on the Tools menu in Tracker Administrator, creates human readable database views and tables of the data in your Tracker project database.
	Database views are <i>virtual tables</i> that are defined as queries based on other tables in the database. The data in these views stays current as the database is updated. This also means that reports based on views can take longer to process than reports on tables, although the views take a negligible amount of storage space because they are composed of references to data rather than the data itself.
	The tables the Assistant creates are of the data as it existed at the time of creation. Unlike the views, the data in these tables is no longer linked to and updated based on the data in the project database.
	The views and tables created by the Assistant are called “human readable” because certain database fields, such as user names and dates, are converted from coded integer numeric values, which are meaningful to the computer, to text or ODBC timestamp format. Additionally, the Assistant creates joins between tables that result in a straightforward layout as opposed to an indirect lookup in the TRKFLD table.

You use third-party tools to create the final custom report.

Using these views and tables and a third-party database tool, such as Microsoft Access or Crystal Reports, you can access the change history and current issue data in one or multiple project databases and make reports of any form you like. Note that a certain amount of proficiency with your database tool will be required to make use of views and tables. See ["Choosing a Database Querying and Reporting Tool" on page 159](#) for more information on required tools.

Required Skills and Knowledge

This section discusses the skills and knowledge you need to have to run the Assistant wisely. Each subsection overviews the information and tells you where to go to learn more.

ODBC Knowledge

If your reporting tool uses ODBC, consult the following resources for more information.

Table 9-1. Resources for learning about ODBC

The primary areas you need to understand are:	To learn more about these topics see:
What a datasource is	A standard computer dictionary, ODBC documentation, or your reporting tool's documentation.
Creating a new datasource	Your reporting tool's documentation or the ODBC documentation.
Connecting to a datasource from your reporting tool	Your reporting tool's documentation.

Database Report Writing Skills

To use the views effectively, you should have a basic understanding of how databases are structured and how data is selected for reporting purposes. The way you select your data is specific to the reporting tool. Some tools may require you to edit SQL directly while others may provide a GUI-based query-defining interface. See your reporting tool's manual for more details. Note that if you already know how to use your reporting tool to create a report showing the contents of the *TRKUSR* table (or any arbitrary table), then you know enough to use the views.

The views can minimize the amount you need to know about the schema.

Familiarity with Your Project Database

Complex tasks, such as a report that answers questions such as “which change requests were submitted by Mary during March 2000 across all projects,” require some knowledge of Tracker's schema so you know what kind of data is available to be reported on. Tracker's schema documentation describes how Tracker structures a database and can be found in the Help for Tracker Administrator. You can access the information via the Help menu in Tracker Administrator or search on *trkadmhw.hlp* in Windows Explorer. Search on “schema” in the index of that help file for more information.

It's also important that you be familiar with the custom fields created for the projects you will be reporting on. You can get a list of your custom fields and their attributes by clicking the Fields icon in Tracker Administrator.



Tracker field names that contain illegal characters such as '#' or '-' need to be modified before they can be used as table or view column headers. As part of its process, the Report Assistant will automatically replace them with '_' in the table or view column headers. If a field name is also a database keyword, “_col” will be appended to it to distinguish it from the keyword. For example, a field named “Select” becomes “Select_col”.

Choosing a Database Querying and Reporting Tool

The Report Assistant itself is really one of a set of tools you will use to create the kind of reports described above. The query and reporting tool you use with the Report Assistant needs to be compatible with your environment and able to access your database.

Tools such as Excel and Crystal Reports can do this. You can also use tools native to your database environment.

Determining the Kinds of Reports You Want to Create

It's best to design the report before running the Assistant, specifying the kind of content you want in each field.

Using the Assistant Under MS SQL Server

This section discusses special considerations for running the Assistant under MS SQL Server.

Database Permissions Required

Verify with your DBA that you have permissions to create views and tables on the target database and read data from all Tracker tables. In addition, you need to be able to:

- Create tables, indexes, and views.
- Drop tables, indexes, and views.
- Grant rights to view the newly created tables or view.

Determine Whether You Should Create Views or Tables

NOTE For the necessary background to make this decision, see the discussion of the difference between views and tables in [“What the Assistant Does for You” on page 156](#).

MS SQL Server is limited to 16 tables in any one view or query. This affects which specific views and tables you can create. See [“Descriptions of the Views and Tables” on page 180](#). When you select fields to convert, the Report Assistant shows the total number of tables included in the view in the *Base Tables* field of the pane for Step 2/5 (see page 169). There are two ways you can compensate for this limitation.

- You can create multiple Converted Fields views, each of which converts only a subset of all possible fields. Since most reports will probably not reference all possible fields, you can use the

Report Assistant to make views that are customized for each report.

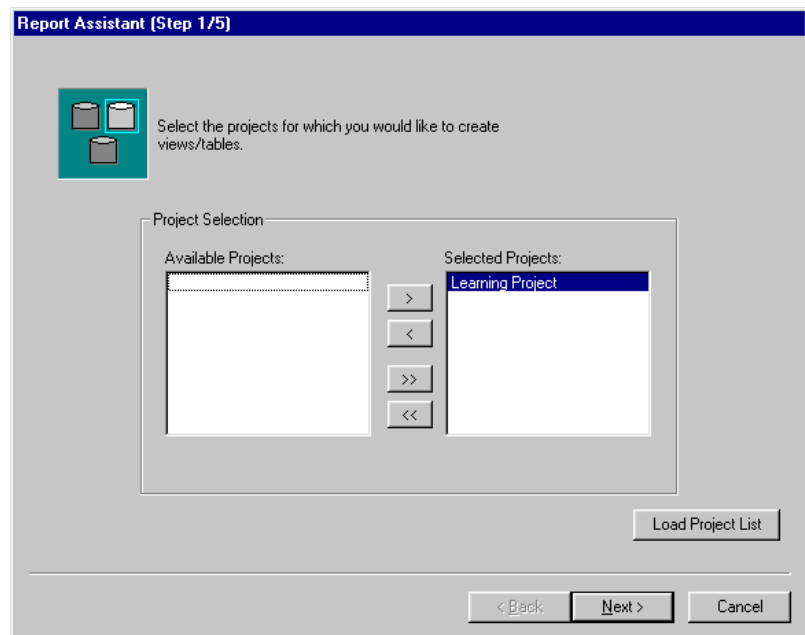
- You can create tables instead of views. This limitation does not apply to tables.

Creating Views and Tables Under MS SQL

Once you have verified that you have the skills and tools required to make use of them, it's time to create the views or tables using the Report Assistant.

To run the Report Assistant:

- 1 Choose Report Assistant from the Tools menu.
- 2 At the Step 1/5 pane, select projects for which you want to create views. If you select multiple projects, you will be able to create views or tables of data in all the selected projects.



- 3 At the Step 2/5 pane, select the views and tables you want to create.

Report Assistant (Step 2/5)

Select the views/tables you want to create. Existing views/tables will be overwritten.

☒ Show Advanced Options

Select the views/tables you want to create

- ☐ View of Change Request Records
- ☒ Table of Change Request Records
- ☐ View of Change History Records

Select the cross project views/tables you want to create

- ☐ View of Change Request Records Across Projects
- ☐ Table of Change Request Records Across Projects
- ☐ Table of Change History Records Across Projects

Advanced options

Please note: project specific views and tables are saved to their respective project databases by default. Cross project views and tables are saved to an Alternate Database, usually TrkMaster.

Alternate location:

- ☐ Put View of Change Requests in Alternate Location
- ☐ Put Table of Change Requests in Alternate Location
- ☐ Put View of Change History in Alternate Location
- ☒ Build Cross Project Change Requests View from Views (not Tables)

You can save the project specific views and tables to the Alternate Database by selecting the appropriate check boxes on the right.

< Back Next > Cancel

By default, cross-project tables are saved to the TRKMASTER database, though you may specify an alternate database to save them to. In addition, cross-project views and tables will contain only those fields where the title and field type are common to all projects. Project-specific views and tables are stored in the project database unless you specify an alternate database to store them in.

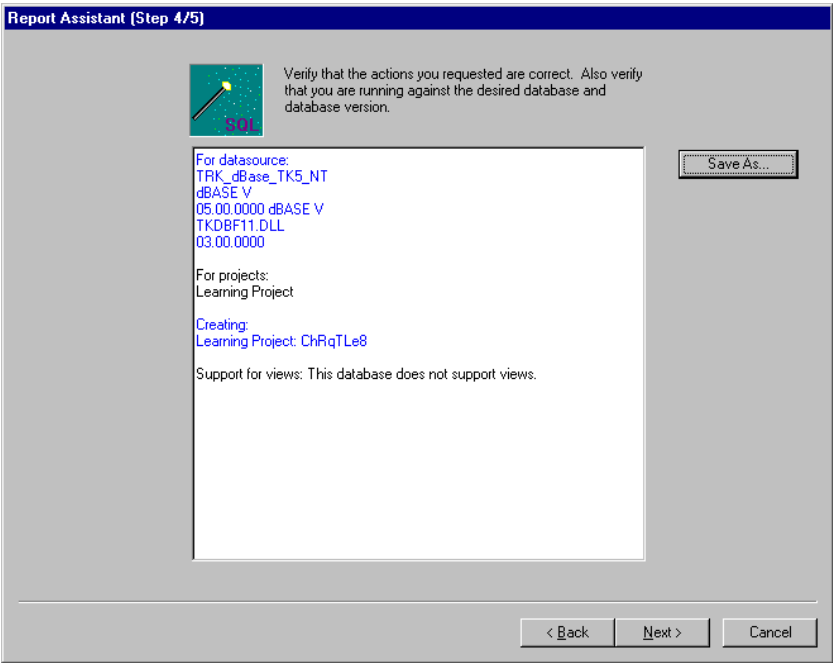
The main Change History view, which is a union of all Change History subviews, cannot be created in MS SQL Server. MS SQL Server does allow UNIONS of views, but the 16 table limitation is still present.

- 4 At the Step 3/5 pane, select the project you want to convert User, Choice, or Date fields for.

You will need to choose to ignore, convert, or leave unconverted each convertible field in each project. The default is to convert all fields. Ignoring the field means that field will not be included in the view or table. Converting the field will, for example, change the numeric value for a user name to the user name. Leaving a field unconverted means that it will appear in the view or table, but the human language equivalent of the numeric identifier will not be displayed.

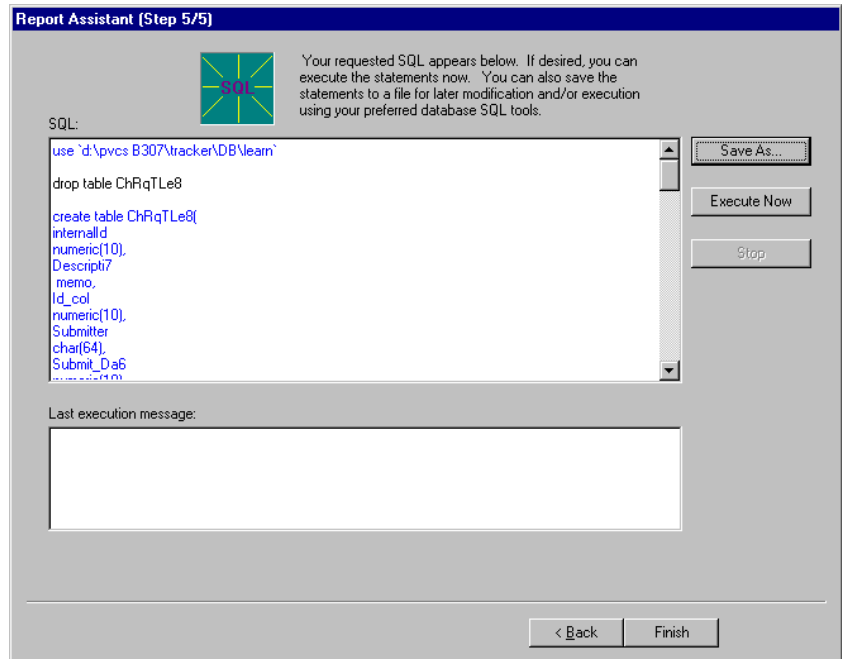
You can determine if you have gone over the 16 table limit by checking the *Base Tables* field in the lower right-hand corner of this pane. Note that Date fields do not affect the number of tables included in the view.

- 5 At the Step 4/5 pane, review the views and tables you have requested to be created for each project. The fields to be converted are listed by project in this scrollable text box.



- 6 If you need to make changes, click the Back button. If you are ready to create views, click the Next button.
- 7 The Report Assistant does not grant specific permissions on the tables and views created. If you want to grant full permissions to everyone for the tables or views created, you can add grant statements at the bottom of the SQL script in the Step 5/5 pane. For example: `grant ALL on DBNAME..TABLENAME to pvcstrkusr` This statement

allows other users to view the views and tables created by this script.



- 8 The SQL statements used to create each view or table are displayed. Review this output for names of tables and views.
- 9 You may now either:
 - Save the script to a text file so you can run it later using the Save As button,
 - Execute the script as it is now using the *Execute* button,
 - Or, click the Finish button to close the Report Assistant.
- 10 Using your reporting tool, connect to the database you want to report on. If you are using ODBC, you may need to create a datasource for each project you wish to report on.
- 11 Create queries and reports with the views and tables as needed.

Using the Assistant Under Sybase

This section discusses special considerations for running the Assistant under Sybase.

Database Permissions Required

Verify with your DBA that you have permissions to create views and tables on the target database and read data from all Tracker tables. In addition, you need to be able to:

- Create tables, indexes, and views.
- Drop tables, indexes, and views.
- Grant rights to view the newly created tables or view

Determine Whether You Should Create Views or Tables

See the discussion of the difference between views and tables in [“What the Assistant Does for You” on page 156](#).

Sybase 11.5 is limited to 192 tables in any one view or query. This affects which specific views and tables you can create. See [“Descriptions of the Views and Tables” on page 180](#). When you select fields to convert, the Report Assistant shows the total number of tables included in the view in the *Base Tables* field of the pane for Step 2/5 (see page [169](#)). There are two ways you can compensate for this limitation.

- You can create multiple Converted Fields views, each of which converts only a subset of all possible fields. Since most reports will probably not reference all possible fields, you can use the

Report Assistant to make views that are customized for each report.

- You can create tables instead of views. This limitation does not apply to tables.

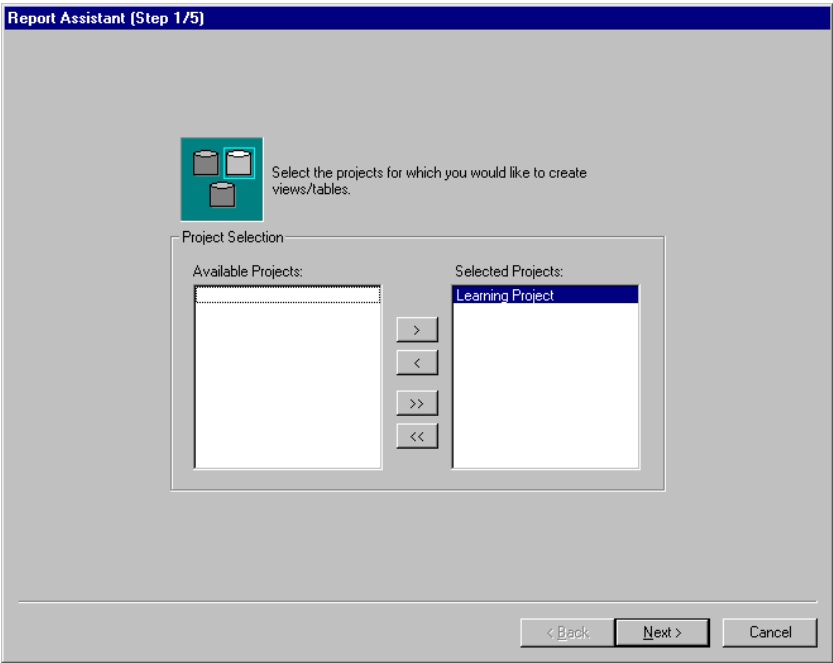
Creating Views and Tables Under Sybase

Once you have verified that you have the skills and tools required to make use of them, it's time to create the views or tables using the Report Assistant.

To run the Report Assistant:

- 1 Choose Report Assistant from the Tools menu.

- 2 At the Step 1/5 pane, select projects for which you want to create views. If you select multiple projects, you will be able to create views or tables of data in all the selected projects.

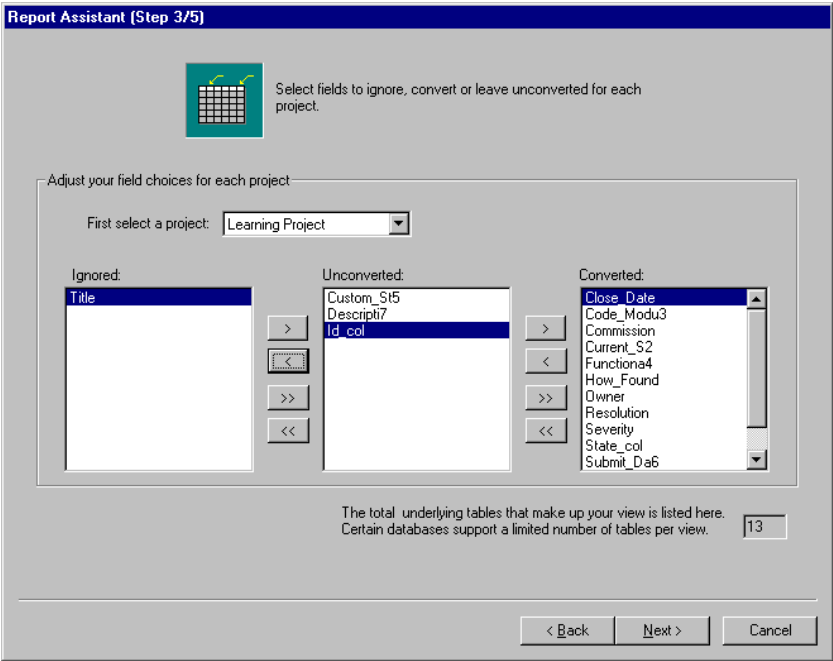


- 3 At the Step 2/5 pane, select the views and tables you want to create.

No cross-project views can be created under Sybase (see [“Determine Whether You Should Create Views or Tables” on page 166](#)). Cross-project tables, however, can be created. By default, cross-project tables are saved to the TRKMASTER database, though you may specify an alternate database to save them to. In addition, cross-project views and tables will contain only those fields where the title and field type are common to all projects. Project-specific views and tables are stored in the project database unless you specify an alternate database to store them in.

The main View of Change History Records, which is a union of all Change History subviews, cannot be created in Sybase. Sybase 11.5 is limited to 192 tables and does not allow a UNION of views. To work around these limitations, you can UNION the sub-views in your reporting tool.

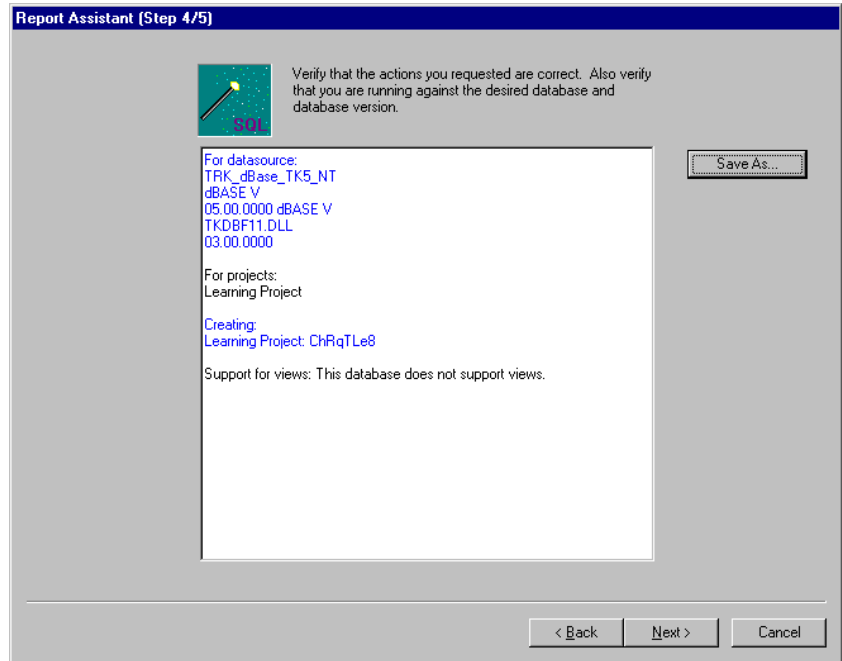
- 4 At the Step 3/5 pane, select the project you want to convert User, Choice, or Date fields for.



You will need to choose to ignore, convert, or leave unconverted each convertible field in each project. The default is to convert all fields. Ignoring the field means it will not be included in the view or table. Converting it will, for example, change the numeric value for a user name to the user name. Leaving a field unconverted means that it will appear in the view or table, but the human language equivalent of the numeric identifier will not be displayed.

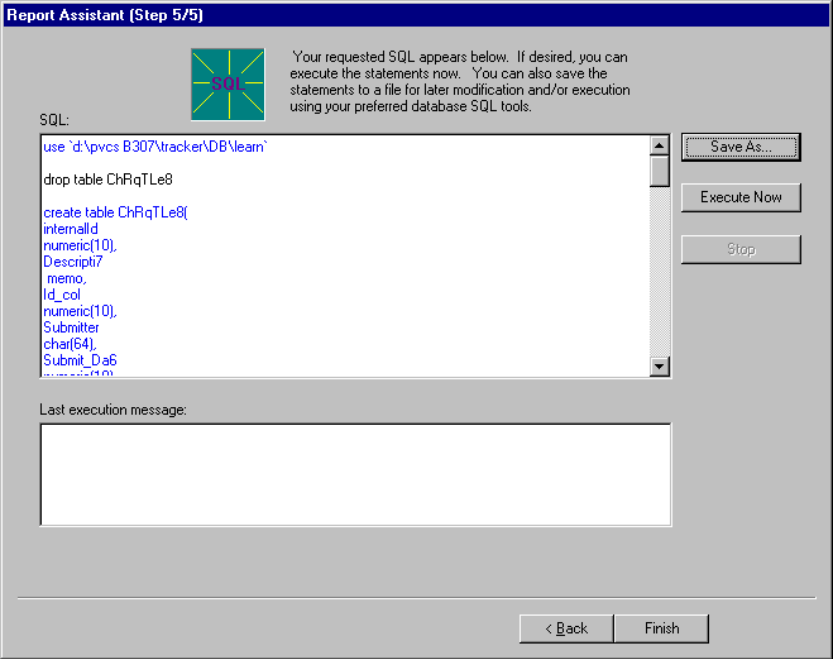
You can determine if you have gone over your maximum number of tables per view by checking the *Base Tables* field in the lower right-hand corner of this pane. Note that Date fields do not affect the number of tables included in the view.

- 5 At the Step 4/5 pane, review the views and tables you have requested to be created for each project. The fields to be converted are listed by project in this scrollable text box.



- 6 If you need to make changes, click the Back button. If you are ready to create views, click the Next button.

- 7 The Report Assistant does not grant specific permissions on the tables and views created. If you want to grant full permissions to everyone for the tables or views created, you can add grant statements at the bottom of the SQL script in the Step 5/5 pane. For example: `grant ALL on DBNAME..TABLENAME to pvcstrkusr` This statement allows other users to view the views and tables created by this script.



- 8 The SQL statements used to create each view or table are displayed. Review this output for names of tables and views.
- 9 You may now either:
- Save the script to a text file so you can run it later using the Save As button,
 - Execute the script as it is now using the *Execute* button,
 - Or, click the Finish button to close the Report Assistant.

- 10 Using your reporting tool, connect to the database you want to report on. If you are using ODBC, you may need to create a datasource for each project you wish to report on.
- 11 Create queries and reports with the views and tables as needed.

Using the Assistant Under Oracle

This section discusses special considerations for running the Assistant under Oracle.

Database Permissions Required

Verify with your DBA that you have permissions to create views and tables on the target database and read data from all Tracker tables. In addition, you need to be able to:

- Create_Any_View
- Drop_View
- Create_Any_Table
- Drop_Table
- Create_Any_Index
- Grant rights to view the newly created tables or view.

Determining Whether You Should Create Views or Tables.

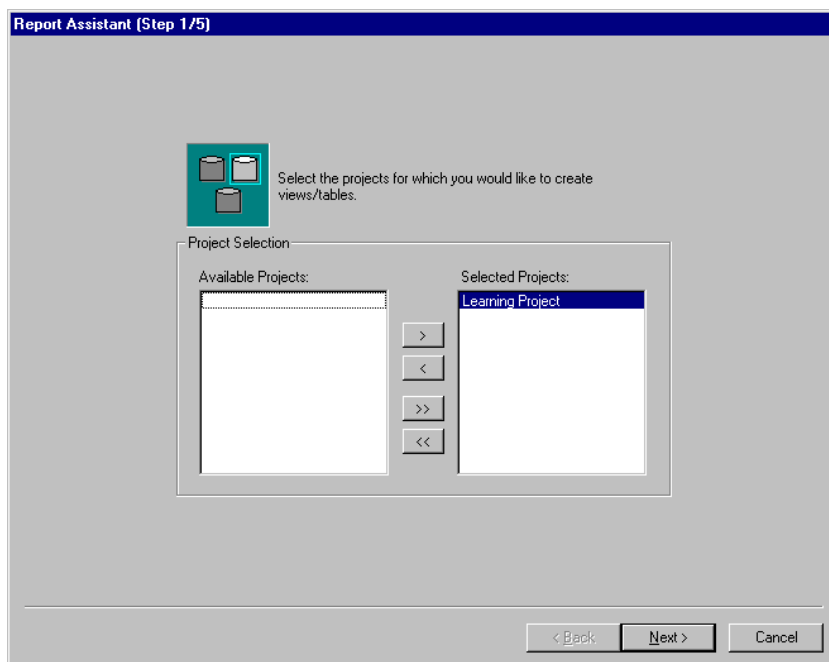
See the discussion of the difference between views and tables in [“What the Assistant Does for You” on page 156](#).

Creating Views and Tables Under Oracle

Once you have verified that you have the skills and tools required to make use of them, it's time to create the views or tables using the Report Assistant.

To run the Report Assistant:

- 1 Choose Report Assistant from the Tools menu.
- 2 At the Step 1/5 pane, select projects for which you want to create views. If you select multiple projects, you will be able to create views or tables of data in all the selected projects.



- 3 At the Step 2/5 pane, select the views and tables you want to create.

Report Assistant (Step 2/5)

Select the views/tables you want to create. Existing views/tables will be overwritten.

☒ Show Advanced Options

Select the views/tables you want to create

- ☐ View of Change Request Records
- ☒ Table of Change Request Records
- ☐ View of Change History Records

Select the cross project views/tables you want to create

- ☐ View of Change Request Records Across Projects
- ☐ Table of Change Request Records Across Projects
- ☐ Table of Change History Records Across Projects

Advanced options

Please note: project specific views and tables are saved to their respective project databases by default. Cross project views and tables are saved to an Alternate Database, usually TrkMaster.

Alternate location:

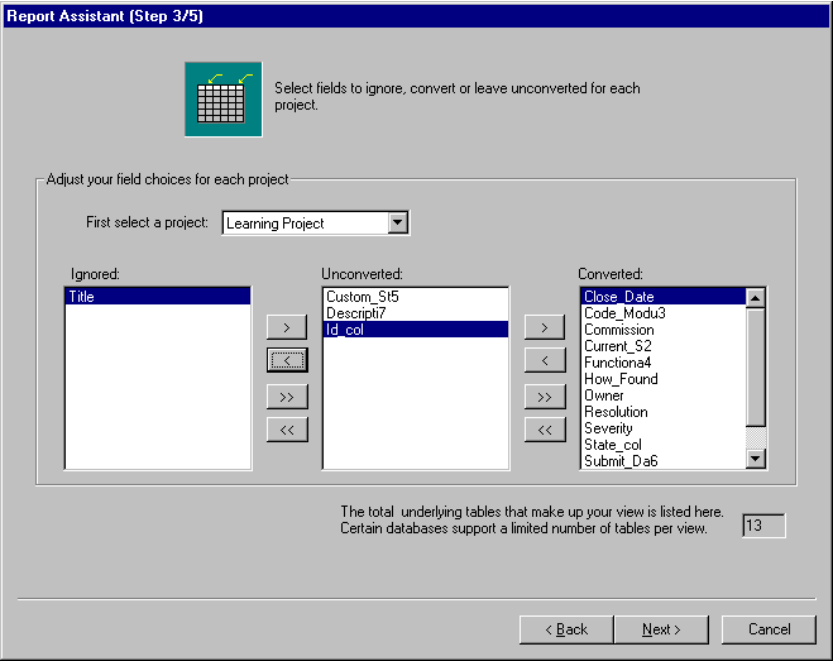
- ☐ Put View of Change Requests in Alternate Location
- ☐ Put Table of Change Requests in Alternate Location
- ☐ Put View of Change History in Alternate Location
- ☒ Build Cross Project Change Requests View from Views (not Tables)

You can save the project specific views and tables to the Alternate Database by selecting the appropriate check boxes on the right.

< Back Next > Cancel

By default, cross-project tables are saved to the TRKMASTER database, though you may specify an alternate database to save them to. In addition, cross-project views and tables will contain only those fields where the title and field type are common to all projects. Project-specific views and tables are stored in the project database unless you specify an alternate database to store them in.

- 4 At the Step 3/5 pane, select the project you want to convert User, Choice, or Date fields for.

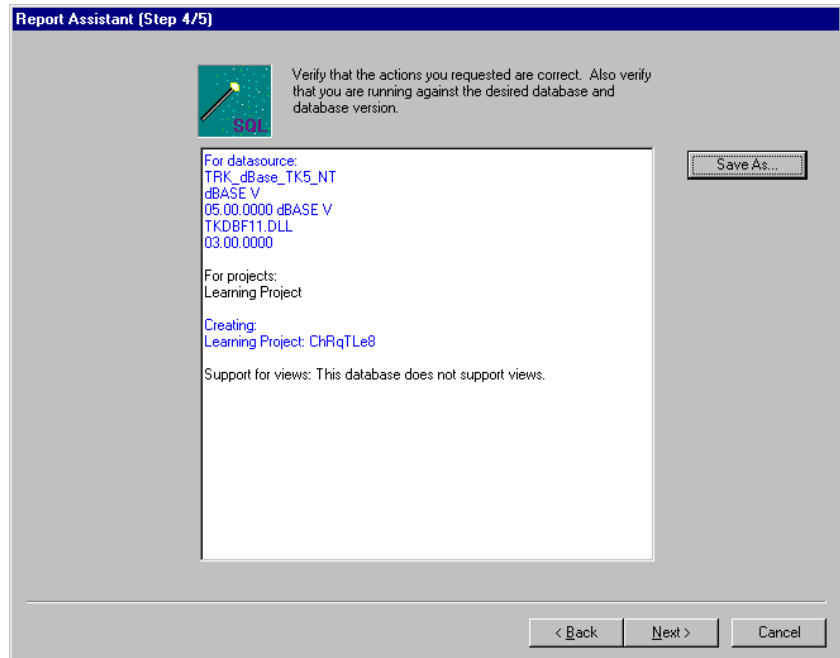


You will need to choose to ignore, convert, or leave unconverted each convertible field in each project. The default is to convert all fields. Ignoring the field means it will not be included in the view or table. Converting it will, for example, change the numeric value for a user name to the user name. Leaving a field unconverted means that it will appear in the view or table, but the human language equivalent of the numeric identifier will not be displayed.

The description field cannot be selected for inclusion in a cross database view or in any table. This is because Tracker creates the description field with a datatype of LONG. The LONG datatype in Oracle cannot be UNION'ed in a cross-project union view. It also means that it cannot be converted: the to_char() function will not accept a long. Also, LONGs cannot be implicitly converted into varchar or chars.

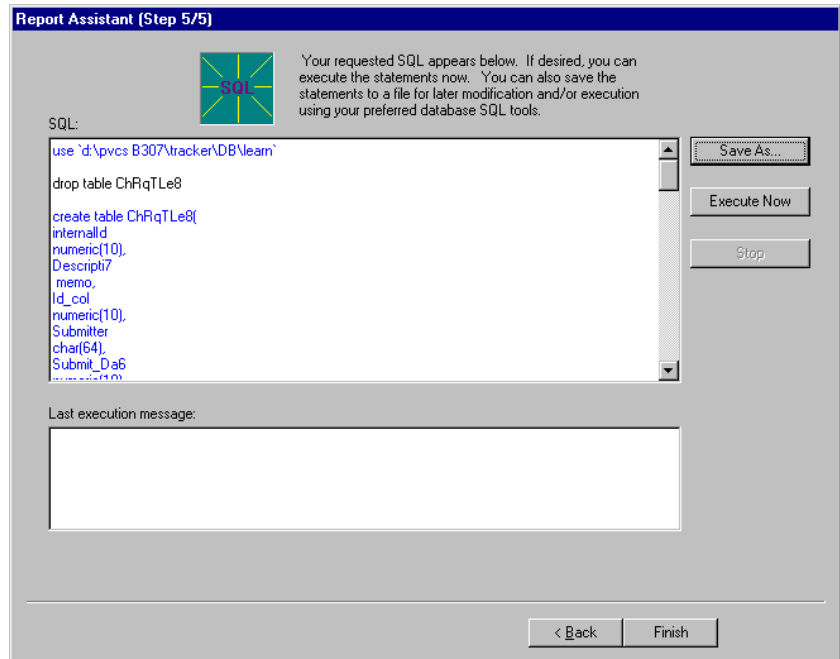
Oracle users can ignore the Base Tables prompt. This prompt is supplied for users with limitations on the number of tables included in queries or views.

- 5 At the Step 4/5 pane, review the views and tables you have requested to be created for each project. The fields to be converted are listed by project in this scrollable text box.



- 6 If you need to make changes, click the Back button. If you are ready to create views, click the Next button.
- 7 The Report Assistant does not grant specific permissions on the tables and views created. If you want to grant full permissions to everyone for the tables or views created, you can add grant statements at the bottom of the SQL script in the Step 5/5 pane. For example: `grant ALL on`

DBNAME.TABLENAME to *pvcstrkusr*. This statement allows other users to view the views and tables created by this script.

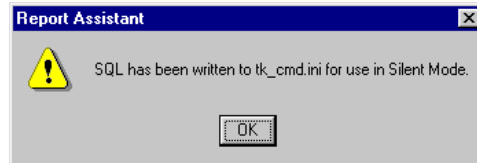


- 8 The SQL statements used to create each view or table are displayed. Review this output for names of tables and views.
- 9 You may now either:
 - Save the script to a text file so you can run it later using the Save As button,
 - Execute the script as it is now using the *Execute* button,
 - Or, click the Finish button to close the Report Assistant.
- 10 Using your reporting tool, connect to the database you want to report on. If you are using ODBC, you may need to create a datasource for each project you wish to report on.
- 11 Create queries and reports with the views and tables as needed.

Creating Project Lists in the Report Assistant

If you find yourself regularly selecting from a large set of projects that appear on Step 1/5 of the Report Assistant, you can use the following procedure to save a set of projects and have it remembered the next time you log in:

- 1 On Step 1/5 of the Report Assistant, click the Load Project List button. A dialog box appears telling you the project list has now been loaded and changes will be saved after pressing Next. Click OK to continue.
- 2 Use the arrow keys to select the projects you wish to work on; then click the Next button.
- 3 Continue with the remaining Steps of the Report Assistant.
- 4 When you come to Step 5/5, click the Finish button. The following dialog box appears:



This indicates that the list of projects has been saved. The next time you use the Report Assistant, your list of projects will appear in the Selected Projects area.

If you later want to modify the saved list of projects, simply click the Load Project List button again and repeat the steps above.

Updating Views or Tables

Each time you add or delete a custom field that you previously included in a view, you will need to run the Report Assistant again to update your views. If you create tables, the data is outdated the moment the table is created, so you may need to update them quite frequently.

Descriptions of the Views and Tables

This section describes what each of the supported views or tables contains. There are two types of views and tables selectable on the Step 2/5 pane of the assistant. Three of the selections are project specific and three are cross-project. Cross-project views and tables will have the same format as shown below with the addition of a “Project” field to distinguish which project each record came from.

Change Request View or Table

This view or table provides issue data reconstructed from the *TRKUSR*, *TRKSCRST*, *TRKSCRSL*, *TRKCH*, and *TRKFLD* tables. Each conversion is essentially a join between tables. Three types of fields are converted as shown in the table below:

Field Type	Located In	Converts from . . .	To . . .
User	trkusr	user ID number	user name
Choice	trkch	choice number	choice label
Date	trkscrsl	number of seconds since January 1, 1970	ODBC time stamp format, GMT time zone

Note that no adjustment is made by the conversion for your local time zone. Only a few standard fields are shown in the unconverted fields example below. The view you create on your database will include your custom fields.

Issue	Title	Submitter	Submit_Date	Resolution
1	Menu popup has line draw char at end of string	13	787351114	4
2	Security, add warning message before deleting user	15	790202314	4
3	Typo in display modify help text	16	790547914	8
4	NIC is not easily accessed w/o opening back	16	790547914	6
5	Can't print error screen until after error is cleared	16	790547914	7
6	Print of error screen is incomplete	14	794176714	4
7	Print disables keyboard access, cancel not recognized	14	794176714	8

The table below shows all convertible fields converted in the Change Request View.

Issue	Title	Submitter	Submit_Date	Resolution
1	Menu popup has line draw char at end of string	Palmer, Arlene	1994-12-13 20:38:34.000	Fixed
2	Security, add warning message before deleting user	Levy, Rich	1995-01-15 20:38:34.000	Fixed
3	Typo in display modify help text	Anders, Kathleen	1995-01-19 20:38:34.000	Postpone
4	NIC is not easily accessed wo opening back	Anders, Kathleen	1995-01-19 20:38:34.000	Do Not Fix

5	Can't print error screen until after error is cleared	Anders, Kathleen	1995-01-19 20:38:34.000	Duplicate
6	Print of error screen is incomplete	Diety, Kevin	1995-03-02 20:38:34.000	Fixed

Change History View or Table

This presents the data from the *TRKCHG* table. It lists all records that show a change to a field value. The values themselves, shown in the *PrevVal* and *NewVal* fields in the example below, are all in string format, including dates and numbers. User, choice, and date fields are all converted as described by the first table under [“Change Request View or Table” on page 180](#).

The main Change History view or table is actually a union of eight sub-views or tables, one for each type of field data: string, user, choice, number, and date. They have names such as:

- ChangeHistory_STRING
- ChangeHistory_USER
- ChangeHistory_CHOICE
- ChangeHistory_NUMBER
- ChangeHistory_notes
- ChangeHistory_files
- ChangeHistory_modules
- ChangeHistory_DATE

They contain the same fields as the main Change History view or tables, but only include the subset of the data appropriate for their field type. If a report only requires change history information from a choice field, for example, it may be more efficient to create ChangeHistory_CHOICE rather than the full Change History view or table.

NOTE The `_DATE` and `_NUMBER` sub-views use timestamp and integer data types for the *PrevVal* and *NewVal* fields, rather than the string data type that the main Change History View uses. This means that if you want to do time related calculations on fields in the PreVal and Next Val columns, you will need to either change the data type for those specific fields from string to timestamp or integer or use the appropriate sub-view.

An example of the main Change History View is shown below.

chgId	Iss_ID	Type	ItemName	PrevVal	NewVal	Change_Date
1	1	USER_FIELD_CHANGED	Owner	<null>	Arlene Palmer	1995-01-02 06:10:05.000
2	1	CHOICE_FIELD_CHANGED	Current State	New	Assigned	1995-01-02 06:12:29.000
412	37	DATE_FIELD_CHANGED	Verify_Date	1995-01-02 06:12:29.000	1995-07-30 09:34:33.000	1995-07-30 09:34:33.000
503	51	STRING_FIELD_CHANGED	Description	You have entered the Learning Project.	Welcome to the Merant Tracker Learning Project! See below for an introduction	1995-07-30 09:34:33.000
83	7	STRING_FIELD_CHANGED	Title	No refresh when minimized	The Config dialog does not refresh when minimized	1995-06-18 17:33:36.000

10 Creating Issue Types and Relationships

In this Chapter

For information about . . .	See page . . .
Overview	186
About Issue Types	186
About Issue Relationships	197

Overview

Who Should Read This Chapter?	This chapter is intended for the Server Administrator, System Administrator, and Project Administrators.
Why Read This Chapter?	This chapter explores how to define multiple issue types and work flows, and create relationships between issue types for improved issue management. Before you can create relationships between issues, you must define issue types and relationship types.

About Issue Types

As a Tracker Administrator, you can group records together by “issue type” - a classification that identifies records of a similar type so that they can be managed differently from other types of records.

For each issue type that you define, you can also set up a work flow, or transition lifecycle, that dictates a sequence of state value passages for continuing to the next phase of the cycle. Further, you can define a specific submit form for each issue type and an update form for each state within the issue type’s lifecycle. For more about issue type work flows, see [“Defining Issue Type Transition Lifecycles” on page 192](#).

Issue types share a common definition of fields among them. When you create or modify an issue type, you can set the valid user-defined fields for that issue type. Then, depending on the issue type, Tracker displays only valid user-defined fields for a particular record on forms and style sheets and when you set group permissions. For example, the field “How Found” would be a valid field for a Defect issue type, but not for a Requirement issue type. If you had not selected “How Found” as a valid field for the Requirement issue type in this example, it would not

appear on the Submit form even though it is listed as a field on that form in Tracker Administrator. System fields are valid for all issue types.

These definitions are described in [“Adding and Modifying Issue Types” on page 188](#). For more about fields, see [“About Fields” on page 210](#).

Any user belonging to a group with Change Issue Type permission can change the issue type in Tracker from one type to another.

Defining Multiple Issue Types

When you create a new issue type for users, you determine the valid fields and icon to display for that issue type. You also assign the state-form values to determine how it will display on the style sheet for the users.

The Issue Type window contains a list of defined issue types for the project that you are currently logged in to.

NOTE If you are a Server Administrator, you have permission to create or edit system issue types. Set the project to <<System>> in the project list to work with system-level issue types. These issue types are available for sharing with user-created issue types so that they can be used across project boundaries. For more about shared issue types, see [“Sharing Issue Types with System Issue Types” on page 193](#).

You can use the Manage Menu on the Issue Types window to do the following:

This command...	Performs this action...
Add Issue Type	Opens the Add Issue Type dialog box, where you can add a new issue type to the selected project.
Modify Issue Type	Opens the Modify Issue Type dialog box, where you can change the name, valid fields, and icons used for the selected issue type.
Delete Issue Type	Removes the selected issue type. You cannot remove the default issue type or any shared issue type.
Recover Issue Type	Restores a previously deleted issue type.
Clone Issue Type	Opens the Add Issue Type dialog box pre-populated with the settings of the selected issue type.
Share Issue Type	Opens the Share Issue Type dialog box, where you can connect the currently selected issue type with a system issue type for cross-project use.
Set As Default	Sets the selected issue type as the default for all new issues in the selected projects. The Current Issue Types list in the Issue Types window shows which issue type is set as the default.
Refresh Now	Refreshes the list of issue types.

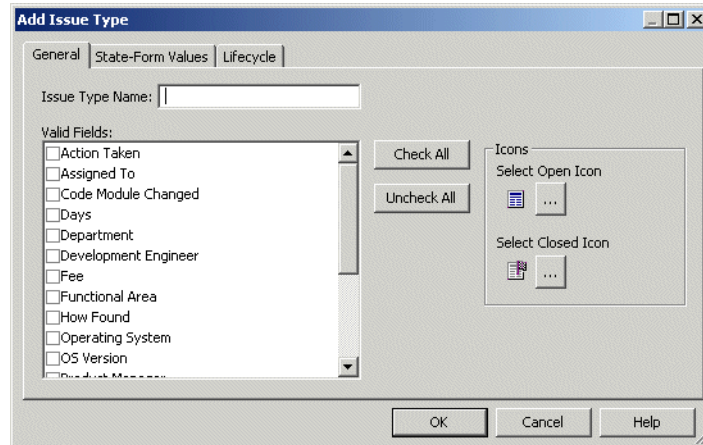
All of these commands except Set As Default are available on the Issue Types toolbar.

Adding and Modifying Issue Types

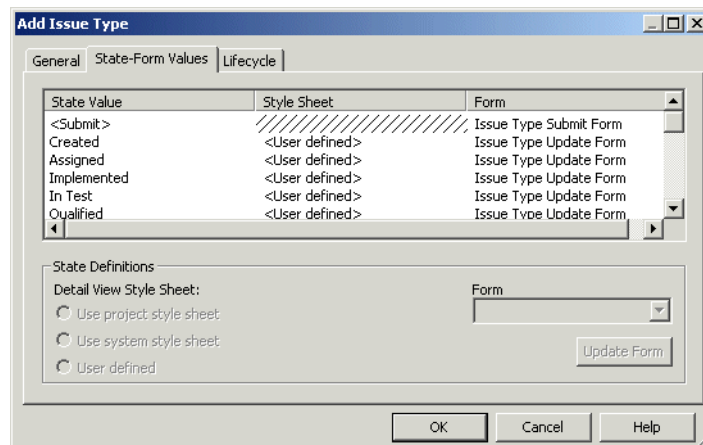
The Project Administrator or Server Administrator can add and modify issue types.

To add an issue type:

- 1 Click the Issue Types button.
- 2 Select the project in which you want to add the issue type.
- 3 Click the Add button. The Add Issue Type dialog box appears.



- 4 Click the General tab to enter the name of the issue type, set valid fields, and choose Open and Closed icons for the issue type.
- 5 Click OK to accept the default style sheet and form values and the Open/Closed icons, or click the State-Form Values tab.



- 6 Select one or more state values. If you choose more than one state value, the Detail View style sheet and form you choose in the State Definitions section are applied to all state values that you selected in this step.

NOTE To add State value choices for setting State-Form and Lifecycle properties, click the Fields button and select the State field label; then click the Modify button.

- 7 Select a Detail View style sheet.

NOTE If a query is set to use a specific style sheet, that style sheet overrides any settings made here.

- 8 Select a submit or update form from the drop-down list. Optionally, you can make changes to the form by clicking the Update Form button to open the Modify Form dialog box.
- 9 Click the Lifecycle tab to configure the default lifecycle state field for the issue type. For more about setting a transition lifecycle for an issue type, see ["Defining Issue Type Transition Lifecycles" on page 192](#).
- 10 Click OK.

To modify an issue type:

- 1 Click the Issue Types button.
- 2 Select the project in which you want to modify the issue type.
- 3 Click the Modify button. The Modify Issue Type dialog box appears.
- 4 Click the General tab to change the name, valid fields, and Open and Closed icons for the issue type.
- 5 Click the State-Form Values tab to change the form used for submitting or updating a record and change the style-sheet used when viewing a record.

- 6 Click the Lifecycle tab to change the transition lifecycle of the issue type.
- 7 Click OK to accept modifications to the selected issue type.

Setting Permissions for Issue Types

Users need to belong to a group with Change Issue Type permission to be able to modify the type of an issue. For example, if you want to allow a user to change the type of an issue from “Defect” to “Requirement”, you need to provide permission to a specific user group.

For more information about setting permissions, see [“Modifying User Group Permissions” on page 135](#).

Permissions for the State Field

To update the State field a user needs update permission on the field. This allows the user to change the State field from a non-final state to another non-final state (or from a non-final state to a final state).

To be able to move the state field from a non-final state to a final state the user needs to have permissions to update the state field and permissions to Enter Final State.

Using Issue Types with Dependent Fields and Transition Relationships

When you set a work flow or transition lifecycle for an issue type, the relationship between the fields is similar to that of transition fields. The field choices for users are restricted to those that you define on the submit and update forms. For more information, see [“Transition Fields” on page 213](#).

You can create dependent fields to further control your lifecycle process. For more information, see [“About Dependent Field Relationships” on page 228](#).

Defining Issue Type Transition Lifecycles

You can define the work flow or transition lifecycle for an issue type when you create it using the Add Issue Type dialog box, or later, using the Modify Issue Type dialog box.

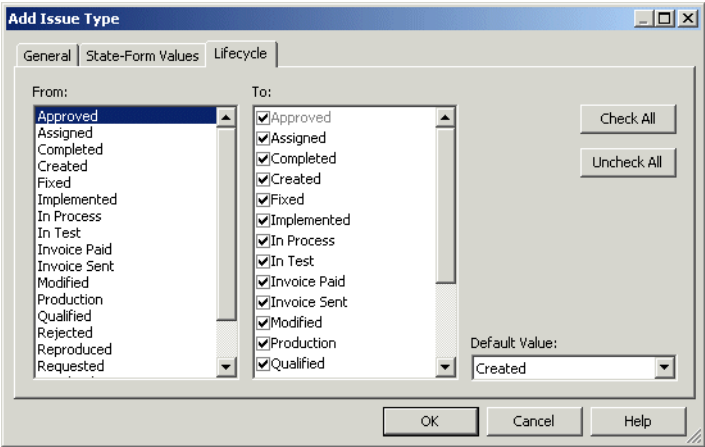
To define a transition lifecycle for an issue type:

- 1 Click the Issue Types button.
- 2 Click the Add button if you are creating a new issue type. The Add Issue Type dialog box appears.

--or--

Click the Modify button if you are creating a work flow for an existing issue type. The Modify Issue Type dialog box appears.

- 3 Select the Lifecycle tab.

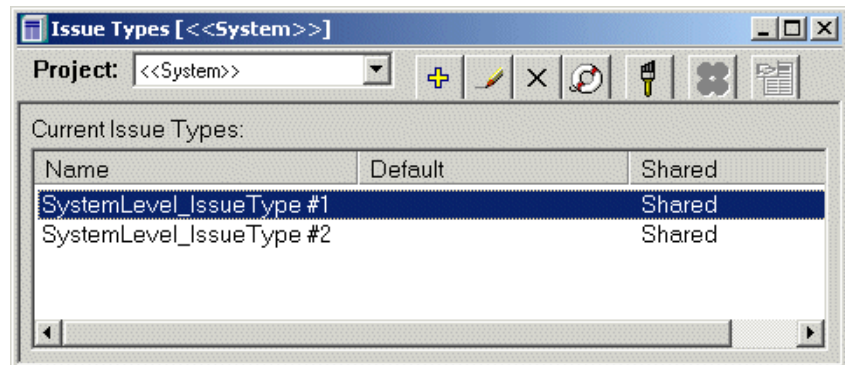


- 4 In the From list, select a state value. Any selections you make in the To list will apply to this selection.
- 5 In the To list, select the state values to be included in the lifecycle for the state selected in the From list.
 - a Click the Check All button to select all state values in the To list.
 - b Click the Uncheck All button to remove all selections of state values in the To list.

For more information on adding and modifying issue types, see [“Adding and Modifying Issue Types” on page 188](#).

Sharing Issue Types with System Issue Types

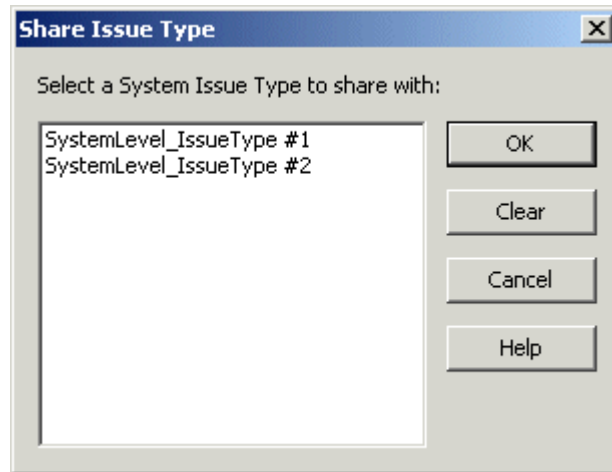
A Server Administrator can create issue types at the system level by setting the Project to <<System>> and clicking the Add button. These system-level issue types do not exist in any project. A system-level issue type controls the issue type to which it is connected.



A Project Administrator can connect an issue type with a system issue type for the purpose of constraining relationships between issue types across projects.

To share an issue type with a system issue type:

- 1 Click the Issue Types button.
- 2 Select an issue type from the Current Issue Types list.
- 3 Click the Share button. The Share Issue Type dialog box appears.

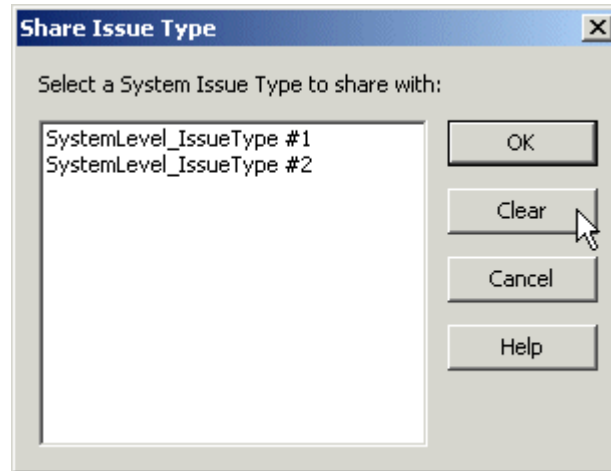


- 4 Select a system issue type from the list and click OK. The issue type takes on the name of the system issue type.

To remove the connection to a shared issue type:

- 1 Click the Issue Types button.
- 2 From the Current Issue Types list, select the name of the shared issue type that you want to disconnect.
- 3 Click the Share button.

- 4 In the Share Issue Types dialog box, click the Clear button.



Deleting Issue Types

You can delete issue types that you no longer need if:

- They are not shared with the system issue type.
- They are not the default issue type.

Delete issue types by selecting it in the Issue Types window and clicking the delete button.

If you delete an issue type that you need to recover, you can use the Recover Issue Type button to return that issue type back to the project.

To recover deleted issue types:

- 1 Click the Recover Issue Type button.
- 2 Click to select the deleted issue type from the Recover Issue Type dialog box.
- 3 Click OK. The recovered issue type appears in the **Current Issue Types** list.



Sample Scenario

Jamal, a new Project Administrator, had recently met with his team and determined that they needed to change the default issue type for their project. They also wanted to define a lifecycle for the issue type. Since Jamal was a new administrator, he called Brad for assistance.

"You will need to set this up for Tracker users in Tracker Administrator," Brad told Jamal. "What issue type do you want to be your default?"

Jamal answered that they were in a planning phase on his project, so they wanted to set the Requirements issue type as the default. Brad instructed Jamal to open the Issue Types window, select the Requirements issue type in the Current Issue Types list, and then choose Set As Default from the Manage menu.

"Good, that was easy," Jamal responded when he saw the Default label move from Defect to Enhancement in the Current Issue Types list.

Next, Brad led Jamal through the process of creating a transition lifecycle for the default issue type.

"First, you need to create all your lifecycle stages for your project as choices in the state field," said Brad. "Do you remember how to do that?"

"In the Fields window, right?"

"Yes, just make sure you select the State field label and modify the Choice/Lifecycle field type," said Brad.

"You set up the lifecycle up when you add or change an issue type," Brad continued. "Since you already have created the issue

type, select the issue type and click the Modify button to go to the Modify Issue Type dialog box. Then select the Lifecycle tab."

Jamal quickly located the Lifecycle tab.

"See the fields you had set in the Configure Choice Field dialog box? You can choose the 'next' states in the 'To' list for every state value you select in the 'From' list," explained Brad.

"OK, I get it. The user will only see the valid choice values for a lifecycle field that I have set in the Configure Choice Field dialog because these values are based on the current value of the lifecycle field. When the user updates the record, they change the value of the lifecycle field. Then, the next user who updates the record will only see the valid choice values for the new value of the lifecycle field. Is that right?"

"That's it," said Brad. Jamal was on his way to creating a robust issue management system using issue types. He would be ready to work with relationship types soon.

(See page [206](#) for a continuation of the scenario.)

About Issue Relationships

Informal and
dependent
classification

As a Tracker administrator, you can create and configure Tracker relationship types to define relationships between allowed issues types. Relationships enable users to have better control and management of issue processing.

A relationship type defines a group of related issues with a set of rules, such as the classification of "[dependent relationship type](#)" or "[informational relationship type](#)". It also designates an issue type as having the "[master-node role](#)" with other designated issue types in the "[detail-node role](#)". Within these roles, you can further define issue relationship rules and restrictions.

Relationships between issue types always involve a master-node role and a detail-node role. For dependent relationships, administrators can constrain the issue types that are allowed on either side of the relationship, for example, making a requirement that the detail role issue must close before the master role issue can close.

There is also a peer role, which is another title for a detail-node role in a relationship. Peer roles are used in sibling relationships, in other words, relationships that share the same parent.

Relationship roles Detail role issues can be set to view their peers within the same type of relationship. Both sides (master-node and detail-node) of dependent relationships have unique role names that are used to describe the relationship to the user within the relationship view. Role titles are shown in the relationship view as a new column.

Relationship constraints When you create a relationship type, you assign it a classification. When you update the relationship type, there are constraints against changing the classification. There is also a closure constraint attribute that constrains the master-node’s ability to enter a final state. If any of the master-node’s children are not currently in a final state, the master-node will not be allowed to enter a final state. If the master-node is currently in a final state, the detail-node will not be allowed to leave a final state.

Default Relationship Types

Tracker Administrator includes predefined relationship types: “parent-child relationship type”, “duplicate relationship type”, and “informational relationship type”. You can add or modify relationship types that are completely customizable for your project’s particular needs.

Understanding Relationship Attributes

There are several attributes that can be applied to relationships. These attributes affect the behavior of a relationship type. Depending on the classification of a relationship type, some of the attributes may not be applicable.

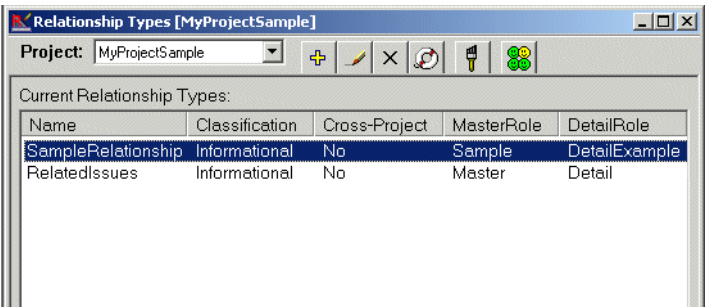
This attribute...	Allows...
Bi-directional Navigation	Navigating or viewing the related record from both sides of the relationship.
Cross-Project Relationships	Records to relate information in multiple projects. This attribute is required for the relationship type in both projects where the relationship is to be created.
Nested Relationships	Records that are current master-nodes of a relationship type can become related as a detail-node of the same relationship type. An example of a nested relationship type would be a parent-child relationship. Relationships without this attribute are constrained to have only one master-node for the specific relationship type.
Viewing Peers	All detail-nodes that are related to a parent-node of the same relationship type to view one another. These records that are not directly related to the current record display the peer role title defined for the detail-node side of the relationship type.
Allowed Types	You to constrain any relationship type to specific issue types by specifying allowed issue types on the master-node, the detail-node or both. In this way, you can prevent certain issue types from being related.

Defining Relationship Types

Relationship types can be constrained to issues within a project, or they can be set up to be visible multiple projects. If a relationship type has cross-project support, it must have an identical configuration across all projects.

When you create a new relationship type for users, you determine whether it is to be used across projects, how it views other relationships and role issues, and the properties of its [master-node role](#) and [detail-node role](#).

The Relationship Types window contains a list of projects that you are currently logged in to. Only system administrators can define cross-project relationship types, but project administrators can define relationship types within a project.



From the Manage menu on the Relationship Types window, you can do the following:

This command...	Performs this action...
Add Relationship Type	Opens the Add Relationship Type dialog box, where you can define a new relationship type for the selected project.
Modify Relationship Type	Opens the Modify Relationship Type dialog box, where you can change rules of the currently selected relationship type.

This command...	Performs this action...
Delete Relationship Type	Removes the currently selected relationship type. Only system administrators can delete system relationship types.
Recover Relationship Type	Restores a previously deleted relationship type.
Copy Relationship Type	Opens the Copy Relationship Type dialog box populated with the settings of the currently selected relationship type.
Share Relationship Type	Opens the Share Relationship Type dialog box where you can assign a shared relationship type alias to use for cross-project relationships.
Refresh Now	Updates the list of relationship types in the Relationship Type window.

Creating Cross-Project Relationship Types

Creating custom cross-project relationship types is a two step process: creating system relationship types and then sharing those types in other projects. When creating cross-project relationship types, ensure the two relationship types in both projects are identical in configuration and that both share the same system type.

When you create system relationship types, you are designing an infrastructure for the records that need to share information in more than one project. Shared system relationship types are what you use to map information from project to another. Shared system relationship types link relationships above the project level.

In addition to setting up the cross-project relationship types, you may want to set up field aliases so that you can view fields for a related record in one project from within another project.

With the infrastructure set up, you can relate records across projects in Tracker.

Creating System Relationship Types for Cross-Project Relationships

System relationship types are used at the system level to relate records between different projects. These system relationship types are the link between projects.

The default relationship types have system relationship types assigned. To share custom relationships, you need to create system-level relationship types to share across projects.

To create system level relationship types:

- 1 Log in to Tracker as the system administrator.
- 2 Open the Relationship Types dialog box.
- 3 Select <<System>> from the **Projects** drop-down list.
- 4 Click the Add button to add a cross-project relationship type. The Add/Modify Shared Relationship Type dialog box appears.
- 5 Enter a name for the relationship in the **Relationship Type Name** field and click OK.

Sharing Relationship Types

Share the relationship types to link the relationships at the system level. When you share system-level relationship types, you set up the map for cross-project relationships.

To share relationship types:

- 1 Select a project from the **Projects** drop-down list.
- 2 Select a custom relationship type that is not already shared from the **Current Relationship Types** list.
- 3 Click the Share Relationship Type button on the Relationship Types window, or select Manage | Share Relationship Type. The Share Relationship dialog box appears.

If the Share Relationship Type button is not active and you want to share this relationship type, you can:

- a Double-click the relationship type name or click the Modify button. The Modify Relationship Type dialog box appears.
 - b Select **Allow relationship to cross project boundaries** on the General tab.
 - c Click OK.
- 4 Select the system relationship type from the Share Relationship Type dialog box.
 - 5 Repeat this process in every project where you want to share the system relationship type to complete the relationship.

Creating Aliases to View Fields in Cross-Project Relationships

Aliases are names used for field types that allow you to view fields in the records across projects. Use aliases to map fields in cross-project relationships to expose a project's fields in other projects.

To create aliases and map them:

- 1 Log in to Tracker as the system administrator.
- 2 Open the Fields dialog box.

- 3 Go to Manage | Shared Aliases.
- 4 Add or modify the aliases by:
 - a Clicking Add to add an alias.
 - b Selecting one in the **Current Aliases** list and clicking Modify to change an existing alias.
- 5 Enter or modify the name in the **Name** field and choose the type of field the alias will represent in the **Type** field and click OK.
- 6 Click OK in the Shared Alias dialog box.
- 7 Go to Manage | Project Field Map to map the alias to the project field. The Project Field Map dialog box appears.
- 8 Map the fields from the Shared Alias field list to the Project Field Mappings list to create field aliases that can be viewed across projects.

Setting Constraints for Cross-Project Relationships

Just as with relationships within a project, cross-project relationships can use constraints to create tight one-to-one relationships. When you constrain types for cross-project relationships, you need to configure the relationship types with the constraints in both projects.

To set constraints for cross-project relationships:

- 1 Select a relationship type to constrain from the **Current Relationship Types** list.
- 2 Double-click, or select Manage | Modify Relationship Type, to open the Modify Relationship Type dialog box.

- 3 Navigate to the Master-node Properties tab. In the Allowed Types group you can choose to:
 - Use loose constraints on the relationship, by selecting **Allow any issue type to be used**. This option allows any new issue types added to be used in this relationship.
 - Use tight constraints on the relationship, by selecting just the issue type(s) in the list to be used in this relationship. The fewer the issue types selected, the more constrained the relationship will be.
- 4 Click OK.
- 5 Repeat the process for the relationship type in the other project that you want to configure with the same constraints.

Deleting Relationship Types

You can delete a relationship types by selecting the relationship in the Relationship Types window and clicking the delete button.

You can also recover that relationship types if you need to get it back.

To recover deleted relationships:

- 1 Click the Recover Relationship Type button.
- 2 Click to select the deleted relationship type from the Recover Relationship Type dialog box.
- 3 Click OK. The recovered relationship type appears in the **Current Relationship Types** list.

Setting Relationship Permissions

Users need to have the Add New Relationship and Delete Existing Relationship permissions to add and delete relationships

in Tracker. These permissions are set for user groups in the Record Tab of the Modify User Group dialog box. For more information about setting user permissions, see [“Modifying User Group Permissions” on page 135](#).

In addition, the System Administrator can expand system-level permissions for Project Administrators to use with relationships. The System Permissions dialog box is available from the Tools menu.



Sample Scenario

Jamal was feeling confident that his group's issue processing would go more smoothly with the new default issue type and lifecycle set up. But he knew that they would soon need more with the growing number of Defect issue types and Enhancement issue types that needed to transition into Requirement issue types.

Jamal decided to create a relationship type to set out some rules by which his project's issue types could be easily managed.

First, Jamal knew that Defect issue types and Requirement issue types should be related to Enhancement issue types because of the likelihood that they would transition at some point to become an Enhancement issue type.

Jamal added a new relationship type and set the classification as Dependent on the General tab of the Add Relationship Type dialog box. He allowed viewing of peer relationships and bi-directional navigation so that project team members could see all related issues in Tracker. Next, he selected the Master Node tab and selected Enhancement as the only issue type to be allowed as a master node.

On the Detail Node tab, Jamal set the Master Type as Enhancement and then selected the option Allow any issue type to be used. This way, his project team could use any issue type including new issue types that were created after the relationship type was added.

Whenever Jamal's project team selected an Enhancement issue type in the summary view in Tracker, they would be able to see all of the issue type's child issues in the relationship view. The child issues' relationship with sibling issues would also be viewable in the Tracker relationship view.

(See page [286](#) for a continuation of the scenario.)

11 Customizing Forms, Fields, and Terms

In this Chapter

For information about . . .	See page . . .
Overview	210
About Fields	210
Adding Fields	214
About Dependent Field Relationships	228
Modifying Fields	231
Deleting Fields	232
Recovering Deleted Fields	233
Copying Fields	234
Preparing Fields for Cloning	235
About Forms	236
Creating Forms	237
Modifying Forms	240
Managing Note Titles	242
Controlling Record Numbering	244
Customizing Terminology	245

Overview

Who Should Read
This Chapter?

This chapter is intended for the Server Administrator *and* Project Administrators.

Why Read This
Chapter?

Tracker Administrator allows you to customize the Tracker interface to suit your development environment. This chapter explains how to:

- Modify standard forms for submitting and updating change requests. You can add new fields and remove fields to customize the forms for your working environment.
- Modify standard fields or define custom fields to customize forms for a particular project.
- Edit note titles. Change system-wide and project-specific titles for sections of notes.
- Customize terminology used in Tracker to suit your development environment.

About Fields

A *field* is a control in an issue in which users enter information or indicate a selection. Tracker enables you to modify the labels for several of the standard fields.

Tracker Standard Fields

The standard fields that ship with Tracker are:

Field Name	Type of Field
Title	String field. It can go up to 80 characters in length.
Description	String field. It can go up to 16384 characters in length.
Submitter	User field. The default setting is the currently logged in user.
ID	Number field.
Submit Date	Date field.
Owner	User field. The default setting is <i>unassigned</i> .
State	Choice field. The default choices are open and closed, but you can remove or rename these or add other choices. Any State choices can be designated as a final state , such as rejected, duplicate, or released.
Closed Date	Date field.

Custom Fields

You can also design a *custom field* for the unique type of information you track at your company, or for a particular project. For example, suppose you use Tracker to track change requests from customers. You could add custom fields in the issue for the customer's name, address, telephone number, and system configuration. Or, you could set up different priority levels to correspond with change requests for different system configurations. You can add custom fields to issues, on a record's submit form, update form, or both.

NOTE After you add or define custom fields, you must assign permissions to it for it to appear on the form.

Data Types Available

Tracker Administrator provides six types of custom fields:

- Choice fields let users choose from values that you specify.
- Date fields let users enter dates within a date range that you specify.
- Number fields let users enter an integer number within a range that you specify.
- String fields let users enter a text string up to a maximum length that you specify.
- Time fields let users enter an amount of time within a range that you specify.
- User fields let users select individual users from one or more groups.

Custom Fields Indexing

You can add custom fields to the list of searchable fields by indexing them. Use this index to sort data by columns allowing Tracker to make quick binary data searches. For example, if you rely heavily on a custom choice field and use it frequently in your queries, you may want to index the field to speed up your query. Use the Index this Field check box in the Configure...Field dialog boxes to index Custom fields.

By default the system fields are indexed and the check box is selected when you create a custom field. Indexing may use more space in your database. If your database has space limitations,

you can choose not to index your Custom fields by clearing the check box.

IMPORTANT! When you index custom field you must re-index before exiting for Tracker. Any changes made to custom field indexes will not take effect until you re-index. Please refer to [“Re-Indexing Database Tables” on page 274](#) for more information.

Transition Fields

Transition fields restrict field choices based on the value of the current field. When you use transition fields in your projects, users can select a sub-set of choice field values, defined by you, on the submit and update forms.

Dependent fields relate values in two different fields; transition fields relate values within a single field. You can use dependent fields with transition fields to further define your process. See [“About Dependent Field Relationships” on page 228](#) for more information on dependent fields.

Limitations on Using Fields

Tracker Administrator enables you to add 120 non-string fields on an update or a submit form (these include date, number, time, choice, and user) and 120 string fields.

NOTE In SQL Server and Sybase, for string fields, one row in the table cannot exceed 1800 bytes.

To add fields on some DBMS types, the Server and Project Administrators require special DBMS permissions. Therefore, make sure your Administrators' DBMS login IDs have been

granted the appropriate permissions. In SQL Server and Sybase, you must be the database owner or sa. In Oracle, you must be a DBA or have been granted the ALTER ANY TABLE system privilege.

NOTE For any of the procedures that follow, you can switch to another project by selecting a different project name from the Project list

Adding Fields

Before you add a field, decide:

- What kind of field to use—choice, date, time, number, string, or user.
- The parameters for the field, such as text labels for choices, minimum and maximum values for numbers, or date ranges for date specifications.

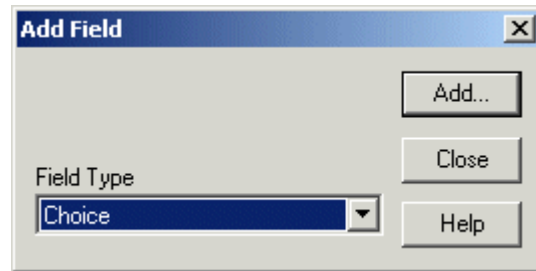
NOTE When you add a new field it defaults to not valid for all issue types. To make the new field visible for an issue type you must add field permissions for the issue type as well as select it as a valid field for the issue type.



To add a custom field:

- 1 Click the Fields button.
- 2 Click the Add button.





- 3 Select the type of field from the **Field Type** list.
- 4 Click the Add button.

Follow any of the following procedures to create custom fields:

For this type of field . . .	See page . . .	For this type of field . . .	See page . . .
Choice	215	Number	223
Date	220	String	225
Time	221	User	226
Transition	219	Dependent	229

Defining Choice Fields

When you define a choice field, you may enter a weight value. The weight value is used to generate statistics in reports regarding, for instance, severity. The weight can be any integer from 0 to 100.

To define a choice field:

- 1 Type the text that will appear in the grouping box for these choices in the Label field.

The 'Configure Choice Field' dialog box is shown. It has a title bar with a close button. The 'Label' field contains the text 'State'. To the right are 'OK', 'Cancel', and 'Help' buttons. The 'Choices' section contains three buttons: 'Add...', 'Modify...', and 'Delete'. Below these is a list box with 'Closed' and 'Open' items, where 'Open' is selected. A 'Drag to reorder' label is to the left of the list box. Below the list box is a checkbox labeled 'Add Choice <<None>>' and a 'Default:' dropdown menu showing 'Open'. The 'Forms' section contains a list box with 'Issue Type Submit Form' and 'Issue Type Update Form', both of which are checked. Below the list box are three radio buttons: 'Not required' (selected), 'Required', and 'Required to enter final state'. At the bottom is a checkbox labeled 'Index this Field' which is checked.

2 To define the choices and their weight values, do the following:

a Choose Add.

The 'Add Field Choice' dialog box is shown. It has a title bar with a close button. The 'Label' field is empty. The 'Weight' field contains the value '0'. There is a checkbox labeled 'Final State' which is unchecked. At the bottom are 'OK', 'Cancel', and 'Help' buttons.

b Type the text that will appear next to this choice in the **Label** field.

- c Enter the weight factor for this choice in the Weight field, where 0 indicates lowest importance and 100 indicates highest importance.
 - d Click OK.
 - e Repeat steps a – d for each choice.
 - f To add <<None>> as a choice, select Add Choice <<None>>.
 - g To designate one of the choices as the default, select it from the Default list.
 - h To modify a choice after you've added it, select the choice and click the Modify button.
 - i To delete a choice after you've added it, select the choice and click the Delete button.
- 3 Select which form(s) to apply this choice field to from the selection of forms in the Forms group.

- 4 (Optional). For each form, if you choose <<None>> as the default, you can choose form specific attributes to indicate when and whether users are required to enter information in this field. To set the attributes for a field on a specific form, highlight the form in the Forms group before selecting the attributes for that field.

Choose . . .	If you require information to be entered . . .
Required	Before the dialog can be processed.
Required to Enter Final State	When the issue is assigned a Closed status or other final state choice in the State field.

If you don't change the default, which is Not Required, the issue can be processed without any value in this field.

NOTE If you select Required, Required to Enter Final State is automatically selected. To associate the form specific attributes with the appropriate form, highlight the form name before selecting the Required options.


- 5 If this is a custom field, the Index this Field check box is selected to index the custom choice field. If you do not want to index this field, clear the check box. If the check box is grayed out, the field is a system field and is already indexed by default. When you add a field to the index or change the existing field index, you must re-index before exiting. Any changes made to the custom field index do not take effect until you have re-indexed.
- 6 Click OK.

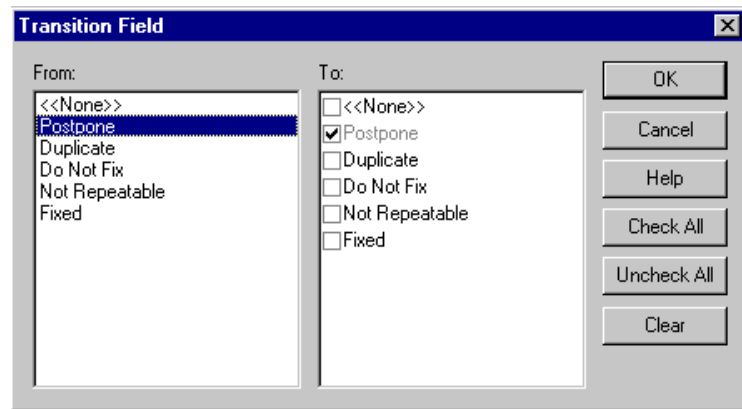
Defining Transition Fields

Values in a transition field are restricted based on the value of the selected choice. You can specify which fields are transition fields and what “next” choices are valid depending on the current value of the field. Refer to [“Transition Fields” on page 213](#) for more information.

To define transition fields:



- 1 Select a choice field in the Fields window and click  or select Manage | Transition Field. The Transition Field Dialog Box appears.



- 2 Select a field value in the From: list to use as the current field value that you want to restrict.
- 3 Select values from the To: list to use as the possible “next” field values.

- 4 Repeat steps 2 and 3 for each value in the From: list to complete setting up your transition field relationships.

Use the **Clear** button to delete the transition relationship.

Defining Date Fields

To define a date field:

- 1 Type the text that will appear next to the date field in the **Label** field.

The screenshot shows the 'Configure Date Field' dialog box. It has a title bar with a close button. The 'Label' field contains the text 'Close Date'. Below it, the 'Range' section has two dropdown menus for 'Minimum' and 'Maximum', both set to 'Unassigned'. The 'Default' section has two radio buttons: 'Current date' and 'Unassigned', with 'Unassigned' selected. The 'Forms' section has two checkboxes: 'Issue Type Submit Form' and 'Issue Type Update Form', both of which are checked. The 'Form specific attributes' section has three radio buttons: 'Not required', 'Required', and 'Required to enter final state', with 'Not required' selected. At the bottom, there is a checkbox labeled 'Index this Field' which is checked. On the right side of the dialog, there are three buttons: 'OK', 'Cancel', and 'Help'.

- 2 To define date range parameters, do the following:
 - a Specify the earliest valid date in the Minimum list.
 - b Specify the latest valid date in the Maximum list.
 - c To designate a default date, select either the Current Date option or the Unassigned option.
- 3 Use options in the Forms group to add this field to the submit form, update form, or both for this record type.

- 4 (Optional). To set the attributes for a field on a specific form, highlight the form in the Forms group before selecting the attributes for that field.

Choose . . .	If you require information to be entered . . .
Required	Before the dialog can be processed.
Required to Enter Final State	When the issue is assigned a Closed status or other final state choice in the State field.

If you don't change the default, which is Not Required, the issue can be processed without any value in this field.

NOTE If you select Required, Required to Enter Final State is automatically selected. To associate the form specific attributes with the appropriate form, highlight the form name before selecting the Required options.

- 5 If this is a custom field, the Index this Field check box is selected to index the custom date field. If the check box is grayed out, the field is a system field and is already indexed by default. When you add a field to the index or change the existing field index, you must re-index before exiting. Any changes made to the custom field index do not take effect until you have re-indexed.
- 6 Click OK.

Defining Time Fields

To define a time field:

- 1 Type the text that will appear next to the time field in the Label field.

Configure Time Field

Label:

Range

Minimum:

Maximum:

Default:

Forms

☒ Issue Type Submit Form

☒ Issue Type Update Form

Form specific attributes:

☒ Not required

☐ Required

☐ Required to enter final state

☒ Index this Field

OK Cancel Help

- 2 To define time range parameters:
 - a Enter the minimum number of minutes in the Minimum field.
 - b Enter the maximum number of minutes in the Maximum field.
 - c Enter the number of minutes that will appear by default in the Default field.
- 3 Use options in the Forms group to add this custom field to the submit form, update form, or both for this record type.

- 4 (Optional). To set the attributes for a field on a specific form, highlight the form in the Forms group before selecting the attributes for that field.

Choose . . .	If you require information to be entered . . .
Required	Before the dialog can be processed.
Required to Enter Final State	When the issue is assigned a Closed status or other final state choice in the State field.

If you don't change the default, which is Not Required, the issue can be processed without any value in this field.

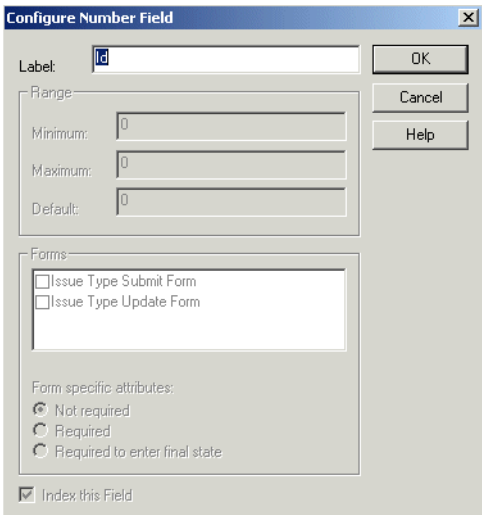
- 5 If this is a custom field, the Index this Field check box is selected to index the custom time field. If the check box is grayed out, the field is a system field and is already indexed by default. When you add a field to the index or change the existing field index, you must re-index before exiting. Any changes made to the custom field index do not take effect until you have re-indexed.
- 6 Click OK.

Defining Number Fields

Number fields are used for ID number specification or to represent a list of build or release numbers. For number fields, Tracker supports only integer numbers.

To define a number field:

- 1 Type the text that will appear next to the number field in the **Label** field.



- 2 To define number range parameters, do the following:
 - a Enter the minimum number in the Minimum field.
 - b Enter the maximum number in the Maximum field.
 - c Enter the number that will appear by default in the Default field.
- 3 Use options in the Forms group to add this field to the submit form, update form, or both for this record type.
- 4 (Optional). To set the attributes for a field on a specific form, highlight the form in the Forms group before selecting the attributes for that field.

Choose . . .	If you require information to be entered . . .
Required	Before the dialog can be processed.
Required to Enter Final State	When the issue is assigned a Closed status in the State field.

If you don't change the default, which is Not Required, the issue can be processed without any value in this field.

NOTE If you select Required, Required to Enter Final State is automatically selected. To associate the form specific attributes with the appropriate form, highlight the form name before selecting the Required options.

- 5 If this is a custom field, the Index this Field check box is selected to index the custom number field. If the check box is grayed out, the field is a system field and is already indexed by default. When you add a field to the index or change the existing field index, you must re-index before exiting. Any changes made to the custom field index do not take effect until you have re-indexed.
- 6 Click OK.

Defining String Fields

To define a string field:

- 1 Type the text that will appear next to the string entry field in the Label field.

The screenshot shows the 'Configure String Field' dialog box. It has a title bar with a close button. Inside, there are three main sections: 'Label' with a text box containing 'Description'; 'Length' with a text box containing '16384'; and 'Forms' with a list box containing two items, 'Issue Type Submit Form' and 'Issue Type Update Form', both of which are checked. Below the list box is a section for 'Form specific attributes' with three radio buttons: 'Not required' (selected), 'Required', and 'Required to enter final state'. At the bottom left is a checkbox labeled 'Index this Field' which is unchecked. On the right side of the dialog are three buttons: 'OK', 'Cancel', and 'Help'.

- 2 Enter the maximum number of characters allowed in this string field in the Length field.

- 3 Use options in the Forms group to add this field to the submit form, update form, or both for this record type.
- 4 (Optional). To set the attributes for a field on a specific form, highlight the form in the Forms group before selecting the attributes for that field.

Choose . . .	If you require information to be entered . . .
Required	Before the dialog can be processed.
Required to Enter Final State	When the issue is assigned a Closed status or other final state choice in the State field.

If you don't change the default, which is Not Required, the issue can be processed without any value in this field.

NOTE If you select Required, Required to Enter Final State is automatically selected. To associate the form specific attributes with the appropriate form, highlight the form name before selecting the Required options.

- 5 If this is a custom field, the Index this Field check box is selected to index the custom string field. If the check box is grayed out, the field is a system field and is already indexed by default. When you add a field to the index or change the existing field index, you must re-index before exiting. Any changes made to the custom field index do not take effect until you have re-indexed.
- 6 Click OK.

Defining User Fields

To define a user field:

- 1 Type the text that will appear next to the user field in the Label field.

Configure Owner Field

Label:

Limit choices to users from user group:

☒ Add Choice <<Unassigned>>

Default: <<Unassigned>>

Forms

- ☒ Issue Type Submit Form 382x381
- ☒ Issue Type Update Form

Form specific attributes:

☒ Not required
☐ Required
☐ Required to enter final state

☒ Index this Field

OK Cancel Help

- 2 Select the user group that contains the user names you want as choices from the Limit Choices to users from user group list.
- 3 To add <<Unassigned>> as a choice, select the Add Choice <<Unassigned>> check box.
- 4 Select the user who will be selected by default from the Default list.
- 5 Use options in the Forms group to add this custom field to the submit form, update form, or both for this record type.
- 6 (Optional). To set the attributes for a field on a specific form, highlight the form in the Forms group before selecting the attributes for that field.

Choose . . .

Required

Required to Enter
Final State

**If you require information to be
entered . . .**

Before the dialog can be processed.

When the issue is assigned a Closed status or other final state choice in the State field.

If you don't change the default, which is Not Required, the issue can be processed without any value in this field.

NOTE If you select Required, Required to Enter Final State is automatically selected. To associate the form specific attributes with the appropriate form, highlight the form name before selecting the Required options.

- 7 If this is a custom field, the Index this Field check box is selected to index the custom user field. If the check box is grayed out, the field is a system field and is already indexed by default. When you add a field to the index or change the existing field index, you must re-index before exiting. Any changes made to the custom field index do not take effect until you have re-indexed.
- 8 Click OK.

About Dependent Field Relationships

Use dependent fields to improve data integrity and simplify data entry. When you configure dependent fields, you restrict available choices in one field based on the selected value of another field.

In a dependent field relationship between two fields, a *child* field is dependent upon a *parent* field. Parent fields are choice fields and child fields are choice or user fields. When a user selects a choice in a parent field on the submit or update form, the list of choices available in the child field shrinks to an appropriate subset of choices.

Use field dependencies to:

- Control your process for submitting and updating issues by restricting the list of available choices.

- Make it easier for submitters to fill out the submit and update forms.
- Improve data integrity.

You can use *token groups* when setting up user fields as dependent. Token groups are smaller collections of cross-functional project teams. Refer to [“Creating Token User Groups” on page 150](#) for information on using and setting up token groups for your projects.

Example of Using Dependent Fields:

Your company’s product runs on a number of supported platforms. The list of vendors and versions has become long and awkward and your users are selecting the wrong options from the list, hence matching “Red Hat Linux” with “XP”.

You configure the *Version* field to be dependent on the *Vendor* field. Now, the *Vendor* field value “MS Windows” has dependent *Version* field values: “XP” and “2000”. The *Vendor* field “Red Hat Linux” has dependent *Version* fields of: “8.0” and “9.0”.

When users enter issues in Tracker against Windows, they only see versions specific to Windows. Likewise, for issues related to Red Hat Linux, they only see versions specific to Red Hat Linux.

Creating Field Dependencies

Dependent fields denote a relationship between two fields where the parent field controls the values in the child field. Child fields are choice and user fields.


You can manage dependent field relationships from within the Fields window to:

- View which field is dependent on another.

- Specify which child field values are allowed for each parent value.
- Set up a default child field value for each parent value.

Dependent relationships don't change how users manage the other aspects of the fields. If a parent field is required, the child field may not be required.

To create field dependencies:

- 1 In the Fields window, click the  button or select Manage | Dependent Fields. The Dependent Field Relationships Dialog Box appears.
- 2 In the Dependent Field Relationships dialog box, select a field from the list that you want to make dependent on another field and click the Child of... button or drag a child to a parent field. The Set Parent for Child Field Dialog Box appears.

NOTE You can use the Map all values by default check box to map all the child values to the selected parent value. This is useful if you need to map nearly all the child values to the parent value. You can select this check box then go to the Map Parent and Child Values dialog box and deselect the one or two values that you do not want mapped.

- 3 In the Set Parent for Child Field dialog box, select the parent field from the Parent Field: list to create a dependency and click OK. The Map Parent and Child Values dialog box appears.
- 4 In the Map Parent and Child Values dialog box, select a parent field value from the Parent Field: list and check the child values you want to assign to that parent field value from the Child Field: list. When a parent field value is bold, it doesn't have any child values assigned to it.
- 5 Select a default child value for the selected parent field in the Default drop-down list. Selecting a default will allow Tracker

to automatically assign a value to the child field on the submit form. If the OK button is grayed out, your default value is not mapped for the parent value.

- 6 Repeat steps 4 and 5 for all the parent fields; then click OK.

NOTE You can let users override the rules by assigning them the “Override field mappings” permission if necessary. Refer to [“Setting Record Permissions” on page 139](#) for more information.

Modifying Fields

You can modify both standard and custom fields.

To modify a field:



- 1 Click the Fields button.

- 2 Do one of the following:



- Select the field you want to modify from the Current Fields list; then click the Modify button.
- Double-click the field you want to modify in the Current Fields list.

- 3 Use the Configure dialog box to modify the field.
- 4 Click OK.
- 5 Repeat steps 2 – 4 for each field you want to modify.

NOTE For more information about the specific fields in the Configure dialog box, see the dialog box information in the online help.

Modifying Field Dependencies

After you establish your field dependencies, you may want to modify the relationships to include more child fields and values or just to adjust it at the request of the project team.

You can modify field dependencies by either:

- Selecting a child field and clicking the Map Values button to change the dependent Values in the child field.
- Dragging a field to a new parent to create a new relationship.
- Dragging a child field to the dependent field relationships tree root to delete the relationship.

Deleting Fields

When you delete a field, you delete the field definition. We recommend that you do not delete standard fields, and most of them cannot be deleted from Tracker. However, it is possible to delete custom fields.

To delete a custom field:



- 1 Click the Fields button.
- 2 Select the custom field you want to delete from the Current Fields list.



- 3 Click the Delete button.
- 4 When you are prompted to confirm the deletion, click Yes.
- 5 Repeat steps 2 – 4 for each custom field you want to delete.

Any field you delete can be recovered throughout the life of the project. For information on recovering fields, see [“Recovering Deleted Fields”](#) in the next section.

NOTE When you delete choice fields with dependent relationships, you delete the supporting relationships as well.

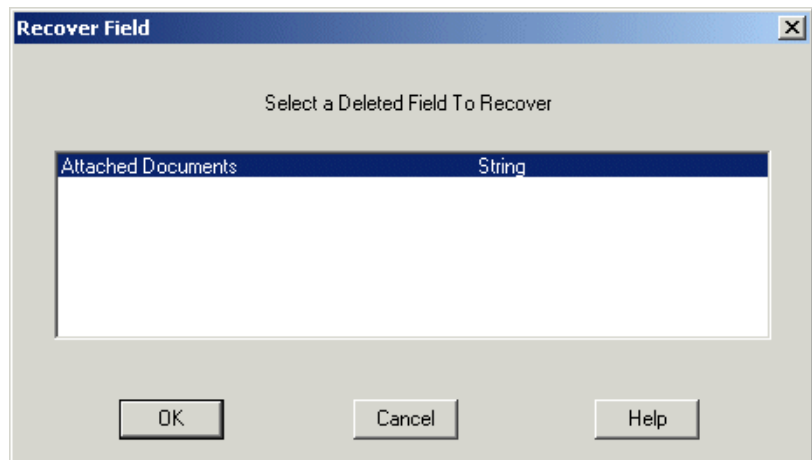
Recovering Deleted Fields

When a field is deleted, it is marked as inaccessible; it is not actually erased from the database. Therefore, any deleted field can be recovered throughout the life of the project.



To recover a deleted field:

- 1 Click the Fields button.
- 2 Click the Recover button.
- 3 Select the name of the field you want to recover; then click OK.



- 4 Repeat steps 2 – 3 for each field you want to recover.

Copying Fields

You can copy fields to other projects so you won't have to add them again.

To copy a field:

- 1 Log in to all the projects you intend to copy custom fields to and from.

NOTE It is recommended, though not required, that you force off the users in the projects you are copying to and from. See [“Controlling Logins” on page 286](#).



- 2 Click the Fields button.
- 3 Select the project you want to copy from the **Current Fields** list.



- 4 Click the Copy button.
- 5 Select the fields you want to copy in the **Fields** list.
- 6 Select the projects to which you want to copy these fields in the **Destination Projects** list.
- 7 Choose Advanced >> to select other options for copying fields. For more information, select Help | Search and enter the keywords *copying fields*.
- 8 Click OK.

Preparing Fields for Cloning

Users can clone records within their Tracker projects. You can choose which fields to have pre-populated in the cloned submit form.

Users need record cloning permissions to clone records. You need to select which user groups will have permissions to clone records. You will also need to decide which fields you will allow to be cloned.

Cloning makes a copy of the record copying over only the specified fields for cloning. After the record is cloned, the users can then update additional fields, add notes, and attach files, to the record. They then submit the cloned record without having to go through the process of submitting a new issue.

To set user group permissions for cloning:

- 1 Set the permission for the user group you want to have permission to clone records.
 - a From the User Groups window, click the Add User Group button.
 - b Click the Record tab and set the **Clone existing record** permission.
 - c Click OK. This sets the cloning permission for the specified user group.

To set up fields for record cloning

- 1 From the Fields window, select Manage | Clone Record Configuration. The Clone Record Configuration dialog box appears.
- 2 Check the project fields that you want to have pre-populate in the cloned submit form.

- Use the Check All and Uncheck All buttons to select all or none of the project fields. If you want to select all the fields but one or two, you can use the Check All button to select all the project fields and deselect the one or two you don't want to clone.
 - Use the Copy Options to check to copy notes and attached files over to the new record when cloning.
- 3 Click OK.

About Forms

Tracker allows you to modify submit and update forms, which allows you to control the information presented to those who submit and update records. You can include or exclude custom and standard fields and control the order in which they appear. You can also create custom forms.

Which Fields on Which Forms?

Selecting fields to appear on forms is dependent on user group permissions. For example, suppose you want a custom choice field for Hardware Configuration on the submit form. If a user group has permission to modify that field, it will appear on the form when the user selects Submit [Issue Type]; otherwise, it will not.

Before you edit forms to assign fields, decide what kinds of submit or update activities will require the broadest possible access to the field. For example, if you have defined a Hardware Configuration choice field, most who submit issues will need access to this field, so you should add it to the submit form. However, it isn't likely to be changed while you are tracking the

issue, so you shouldn't need to have this field on the update form.



NOTE To refresh the list of forms to reflect changes made since you opened a task window, click the Refresh button.

Creating Forms

If you have a project that requires fields different from the defaults and modifying the form won't do, you can add a new form to your project. The following procedure helps you create forms in addition to the standard submit and update forms that Tracker comes with.



To create a custom form:

- 1 Click the Forms button.



- 2 Click the Add button. The Add New Form dialog box appears.

- 3 Give the form a unique name.
- 4 Choose from the field names in the Available Fields box to build a list of fields for this form in the Selected Fields box and order them using the Move Up and Move Down buttons.
- 5 In the Help URL area, you can specify the name of a file or web page that gives custom help on your specific processes.

NOTE This help file will replace Merant’s help on the Submit or Update form. You might want to copy that help to the bottom of your custom help web page to make it available

to your users.

- 6 In the Layout Options area, select the appropriate choice:

Select this...	For this result...
Use auto-field layout	Tracker automatically lays out the fields for you. This is the default setting.
Use detail view style sheet	Tracker uses the style sheet selected by the user for the detail view. This allows the user to specify the style sheet for the Submit or Update form.
Use project style sheet	Tracker uses the project style sheet.
Use system style sheet	Tracker uses the system style sheet.

- 7 Change any form specific attributes you feel need to be changed for this form. Use the following as a guideline:

Choose . . .	If you require information to be entered . . .
Required	Before the dialog can be processed.
Required to Enter Final State	When the issue is assigned a Closed status or other final state choice in the State field.

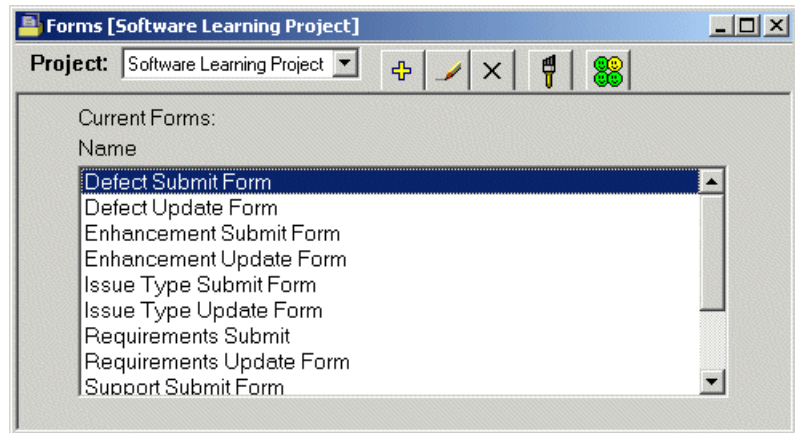
NOTE If you select Required, Required to Enter Final State is automatically selected.

Modifying Forms



To modify a form:

- 1 Click the Forms button.
- 2 Do one of the following:
 - Select the form you want to edit from the Current Forms list; then click the Modify button.
 - Double-click the form you want to edit in the Current Forms list.



- 3 To add a field to the form, select it in the Available Fields list and click the Add button.

Modify Form

Form name:

Available fields:

Selected fields:

Buttons: Add ->, <- Remove, Move Up, Move Down

Help URL:

(Leave blank for default help)

Layout Options:

- ☒ Use auto-field layout
- ☐ Use detail view style sheet
- ☐ Use project style sheet
- ☐ Use system style sheet

Form specific attributes:

- ☐ Not required
- ☐ Required
- ☐ Required at close

Buttons: OK, Cancel, Help

- 4 In the Help URL area, you can specify the name of a file or web page that gives custom help on your specific processes.

NOTE This help file will replace Merant's help on the Submit or Update form. You might want to copy that help to the bottom of your custom help web page to make it available to your users.

- 5 To remove a field from the form, select it in the Selected Fields list and click the Remove button.
- 6 To require users to enter information in a field, select the field in the Selected Fields list; then select the Required option in the Requirements group.

- 7 To require users to enter information in a field before closing the record, select the field in the Selected Fields list; then select the Required to Enter Final State option.
- 8 In the Layout Options area, select the appropriate choice:

Select this...	For this result...
Use auto-field layout	Tracker automatically lays out the fields for you. This is the default setting.
Use detail view style sheet	Tracker uses the style sheet selected by the user for the detail view. This allows the user to specify the style sheet for the Submit or Update form.
Use project style sheet	Tracker uses the project style sheet.
Use system style sheet	Tracker uses the system style sheet.

- 9 Click OK.
- 10 Repeat steps 2 – 7 for each form you want to edit.

Managing Note Titles

Tracker provides system-wide and project-specific titles for notes. You can create the note titles shown in the New Notes dialog box when a user updates an issue. Users can also create unique note titles when entering notes in the Add Note dialog box as well.

Adding Note Titles

To add a note title:



- 1 Click the Notes button.
- 2 Do one of the following:
 - To add a note title on a system-wide basis, select <<System>> from the Project list.
 - To add a note title for a particular project, select the project from the Project list.
- 3 Click the Add button.
- 4 Type a new note title in the Note Title field.
- 5 Click OK.
- 6 Repeat steps 2 – 5 for each note title you want to add.



Modifying Note Titles

To modify a note title:



- 1 Click the Notes button.
- 2 Do one of the following:
 - Select the note title you want to modify from the Current Note Titles list; then click the Modify button.
 - Double-click the note title in the Current Note Titles list.
- 3 Type a new note title in the Note Title field.
- 4 Click OK.
- 5 Repeat steps 2 – 4 for each note title you want to modify.

Deleting Note Titles

To delete a note title:



- 1 Click the Notes button.
- 2 Select the note title you want to delete.
- 3 Click Delete.
- 4 When you are prompted to confirm the deletion, click Yes.
- 5 Repeat steps 2 – 4 for each note title you want to delete.

Controlling Record Numbering

Issues are numbered sequentially (starting with 1) as they are added to the project database. The number is referred to as the record ID. You may find it useful to skip some numbers and set Tracker to number new records starting at a different point.

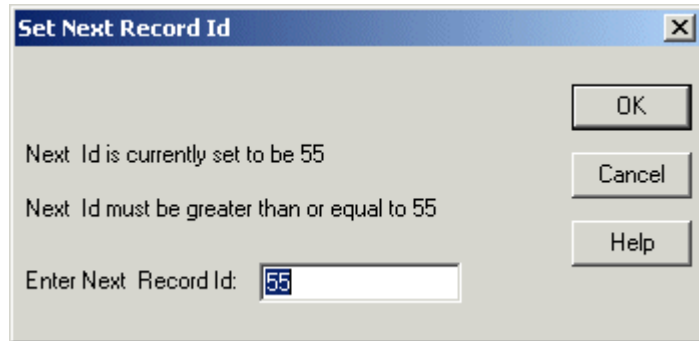
For example, you can separate issue records into ranges for different releases of your software. If you had 230 issues with the first release, you could set the next issue record ID to 250. Then all issues associated with the second release would be numbered 250 and higher.

To set the next record ID:



- 1 Click the Projects button.
- 2 Select the project for which you want to set the next record ID in the Current Projects list.

3 Select Manage | Reset Record ID.



4 Select the type of record for which you want to reset the record ID from the Select Record Type list—either issue or Time.

5 Enter the record ID to be used for the next record added in the Enter Next Record ID field.

NOTE The next record ID must be greater than or equal to the current record ID.

6 Click OK.

Customizing Terminology

Tracker allows you to substitute a different term for the standard terms listed below:

- In Tray
- Issue
- Open
- Closed

- Submit
- Update

When you substitute a different term, Tracker uses that term wherever the standard term would normally appear. For example, you can change the term Issue, and your new term will be displayed wherever Issue was displayed in menu items, dialog boxes, and standard report titles.

You can specify an eight-character prefix, a short name, and a long name for any of the terms above. Tracker automatically uses the short name when the long name won't fit.

NOTE For the procedures that follow, you can switch to another project by selecting a different project name from the Project list:



To refresh the list of terms to reflect changes made since you opened a task window, click the Refresh button.

Customizing Terms

To customize a term:

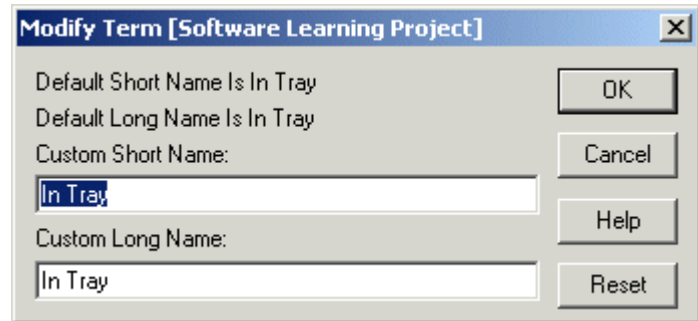


- 1 Click the Terms button.
- 2 Do one of the following:



- Select the term you want to customize from the Current Terms list; then click the Modify button.

- Double-click the term you want to customize in the Current Terms list.



- 3 Enter an eight-character prefix in the Custom Prefix field, if applicable.
- 4 Enter a short name in the Custom Short Name field.
- 5 Enter a long name in the Custom Long Name field.
- 6 Click OK.

NOTE To restore a field to its original Tracker term, select Reset.

Resetting Terms to Their Default Value

To reset a term to its default value:

- 1 Click the Terms button.
- 2 Do one of the following:
 - Select the term you want to reset from the Current Terms list; then click the Reset button.
 - Select the term you want to reset from the Current Terms list; then select Manage | Reset Term.



- 3 Repeat step 2 for each term you want to reset.

Custom Terms and the Tracker Web Client

In Tracker 6.5 and later, the Tracker web client is able to use the custom terms created by the Tracker desktop client, with the following exceptions:

- Terms in help pages and button text
- Terms in any other HTML pages that are not dynamically generated

12 Integrating Tracker with Source Control

In this Chapter

For information about . . .	See page . . .
Overview	250
Benefits of Integrating Tracker and Source Control	251
About Version Manager Configuration Files	251
About Change Text	253
Customizing Source Control Setup	254

Overview

Who Should Read
This Chapter?

This chapter is intended for Server Administrators.

Why Read This
Chapter?

Integrating Tracker with source control enables you to join source control revision histories with the system change request (issue) history that is part of your issue database. In this chapter, you'll learn:

- The efficiency and time-saving benefits of integrating Tracker with source control
- How to use configuration files to control Version Manager's behavior
- How to change the Tracker setup using the Source Control Setup dialog box so that Tracker and your source control provider can exchange project-specific data

Before You Read this Chapter

We recommend that you refer to the *Merant Tracker User Guide* to read about:

- The conceptual model for the integration of Tracker to source control
- How to use the source control features of Tracker

You should also have a working knowledge of your source control provider before performing the tasks described in this chapter. For more information on using Version Manager, see the *Merant Version Manager User's Guide*.

Benefits of Integrating Tracker and Source Control

Using integrated products benefits your development and maintenance efforts in several ways:

- “Fixed” issues often need to be reopened. You can point at an issue and instantly zero in on code changes that are part of previous investigations.
- You can pinpoint specific modules that are candidates for re-engineering.
- You can use the traceability between source code changes and issues in complying with internally or externally imposed audits of the change process.
- “Fixed” modules often introduce new bugs. You can point at a broken code module and instantly identify recent issue activity.
- One-step change documentation encourages use, saves developer time, and reduces errors. You don’t need to manually link code changes with issue records; Tracker does it for you.

About Version Manager Configuration Files

A Version Manager configuration file is a text file containing keywords called *directives*. Directives define the configuration choices that control how Version Manager operates. For example, the configuration file contains directives that specify the location of archives and the default behavior for checking files in and out.

You can create a project configuration file or a master configuration file depending on the behavior you want to control.

This type of configuration file ...	Enables you to fine-tune Version Manager for ...
project	Individual projects. Refer to “Customizing Source Control Setup” on page 254 to specify the configuration files that you want Version Manager to use.
master	A group of projects. You need to embed the master configuration file into the Tracker Version Manager *.DLLs for it to work with your projects. Refer to “Embedding a Master Configuration File” on page 252 to do this.

NOTE For more information on creating configuration files, see the *Merant Version Manager User’s Guide*.

Embedding a Master Configuration File

Use this procedure if you want all of your Version Manager projects to share the same directives.

To embed a configuration file:

For this program ...	Go to a DOS prompt and change to this directory ...	Then type this command and press Enter ...
Tracker	<code>drive:\path\nt</code>	<code>trkvcfg -cmaster.cfg trkvcn.dll</code>
Notify	<code>drive:\path\nt</code>	<code>trkvcfg -cmaster.cfg tr2vcn.dll</code>

Where *drive:\path* is the location of your Tracker or Notify installation, and *master.cfg* is the name of your Version Manager configuration file.

About Change Text

When a user checks in files from within Tracker, Tracker passes change text that has been augmented with issue information and a version label to your source control provider. After your source control provider checks in the files, it returns revision information to Tracker. Tracker adds the revision information to the change text that is associated with the issue. You can configure a template for both the change text and the version label in the Source Control Setup dialog box.

How Change Text is Defined

You control the content and format of the information added to the change text via the Source Control Setup dialog box. The Source Control Setup dialog box is used to create templates for the default source control and Tracker change text and to control which configuration files are used by Version Manager. Using customized templates, you can automatically augment change text and version labels with information on who made the change, which problems and code modules were involved, and the date and time of the change.

To create customized templates, see the next section.

Customizing Source Control Setup

Use this procedure to create a custom template for reporting issue changes to source control.



To customize source control setup, follow these steps:

- 1 From the Tracker Administrator tool, click the Projects button.
- 2 Select the project you want to integrate from the **Current Projects** list.
- 3 Select Manage | Source Control Setup. The Source Control Setup dialog box appears.

Source Control Setup

Default Change DescriptionTemplate:

Project: \$project
Record(s): \$id
\$title
\$change

Default Version Label

Issue-\$id

Merant Version Manager Configuration Files

Add...
Delete

OK
Cancel
Help

Codes

- \$change
- \$id
- \$ownid
- \$owner
- \$project
- \$title

- 4 In the **Default Change Description Template** field, enter text and/or codes. The codes expand to source control and issue information in each Tracker module description and source control change description.

The following codes are available for use in change descriptions and version labels:

This code . . .	Expands to . . .
\$change	The change text
\$id	Issue ID number
\$ownid	User ID of the issue owner
\$owner	Name of the issue owner
\$project	Name of the current project
\$title	Title of the issue

- 5 In the **Default Version Label** field, enter text and/or codes. The codes expand to source control and issue information. When a user checks in a module, the Label field is populated with text derived from the text and codes you enter here.

NOTE The user can override the default and specify their own label or apply no label at all.

- 6 If Version Manager is your source control provider, you can specify a Version Manager configuration file:
 - To add a configuration file, click the Add button.
 - To remove a configuration file, select it and click the Delete button.
- 7 Click OK.

13 Setting Up Notify

In this Chapter

This section...

[About Notify](#)

[Using Notify](#)

See page...

[258](#)

[262](#)

About Notify

Notify is a Merant Tracker feature that allows users to learn about additions and changes to a Tracker project database through e-mail notification.

Notify runs as a Windows Service and is installed and managed by the Windows Service Manager. Tracker Notify Service is configured for automatic startup, so it starts whenever the system is started whether a user is logged in or not. When a user logs in to the system, the user can determine whether Notify is running by checking the status of the service in the Windows Service Manager.

NOTE Notify Service will not run if you do not have an e-mail server configured. See [Configuring Your Mail System](#) on [page 266](#).

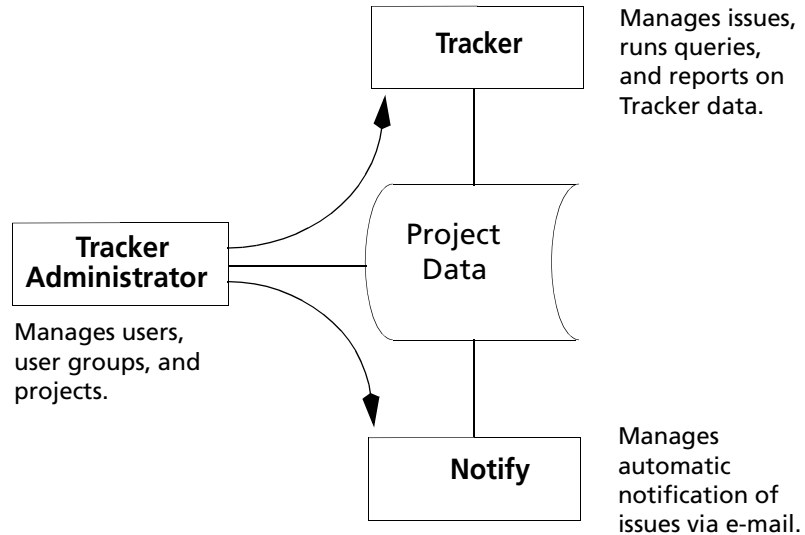
Notify Concepts

When new records are added or fields in existing records are updated, Tracker can generate an automatic notification about the change.

The Server or Project Administrator sets notification options in Tracker Administrator that decide if and how notification occurs for each user and user group. A user's relationship to a particular issue and the user's function in the organization can help the Administrator decide how the user's notifications should be set up.

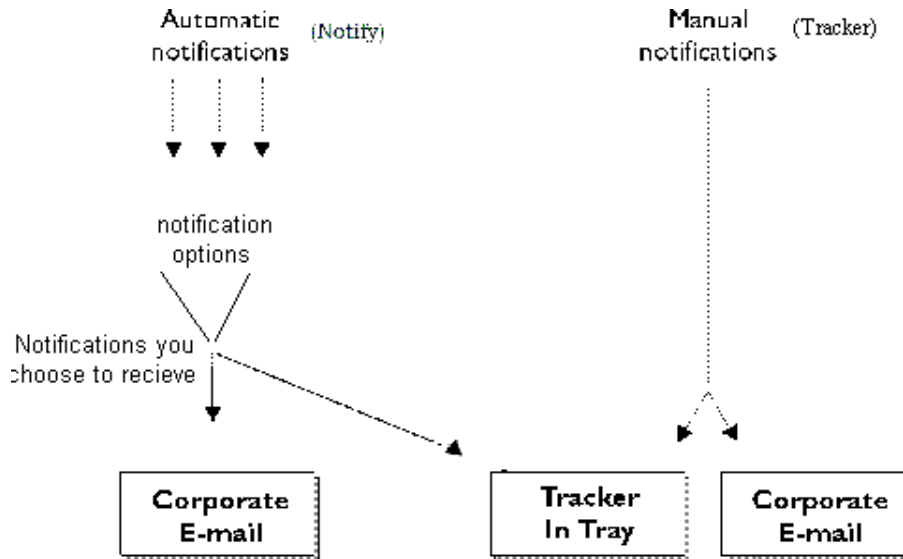
Additionally, individual users may configure their own notification rules, if the Administrator allows this.

The following diagram summarizes the relationships among Notify, Tracker, and Tracker Administrator.



If users choose to be notified of changes to records via the Tracker In Tray, they must log in to the Tracker project to receive those messages. This is effective if they are actively involved in the project and log in to that project's database frequently. If this is not the case, they might prefer to have automatic notifications sent to them via e-mail using Notify. In this way, they can route notifications from multiple projects to a single destination. Notifications sent via e-mail appear in the inbox of a user's mail system. Users can also generate manual notifications through Tracker and send them via the Tracker In Tray or the corporate e-mail system.

The following graphic illustrates the Tracker notification process.



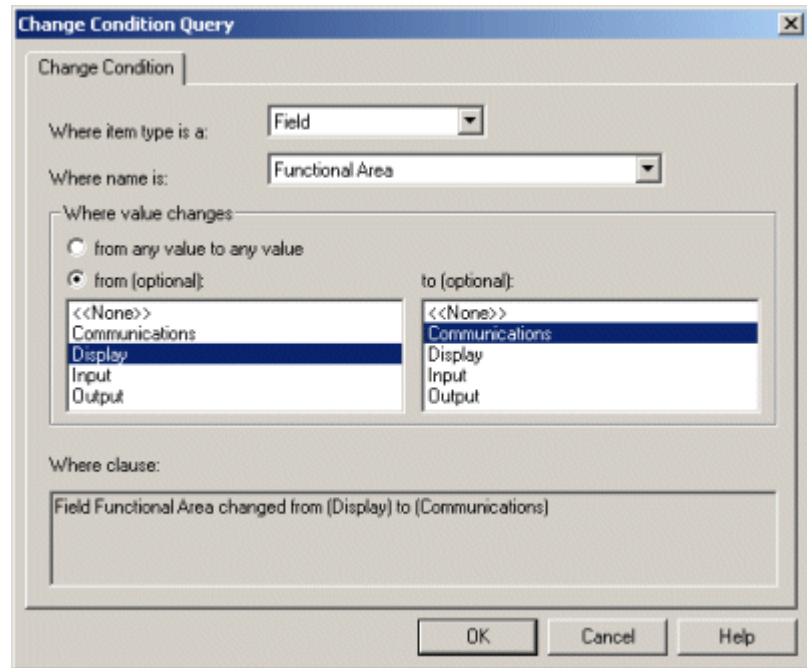
Notify Options

Notification options (also called rules) help you keep track of record additions and field changes. Each user can set these options for issues logged in Tracker. Users are notified based on the notification rules set for their user ID and for any user group they belong to. See [“Setting Up Notify Options for a User” on page 112](#) for more information on setting up notification options.

Conditional Notify Options

You can set conditional notification options for *from* and *to* field values on choice and user fields. When the options are set, Notify alerts users that specified information has changed from one value to another value. These settings are in the Change Condition dialog box (which appears when you click the When

button in the Define Notify Rule dialog box) in Tracker Administrator and Tracker.



Complex Notification Rules

You can also create complex rules using Boolean logic (incorporating *and*, *or*, *not*, and parentheses) in the Query Properties dialog box (Tools | Notify Rules | Add | Notification Rule | Where button). For example, you might want to be notified when you become the owner of an issue only if the Priority field is set to critical.

Using Notify

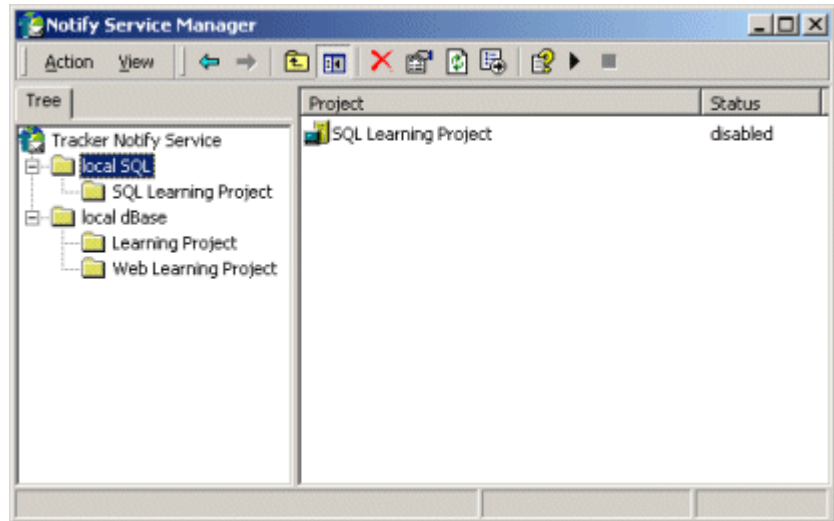
Starting Notify

Notify runs as a Windows Service. It is installed with and managed by the Windows Service Manager. It can be configured for Automatic startup, so it starts whenever the system is started regardless of whether or not users are logged in.

You can determine if Notify is running by checking the status of the service in the Windows Service Manager. Go to **Control Panel | Administrative Tools | Services**, locate **Tracker Notify Service**, and look at the Status column. To start the service, double-click **Tracker Notify Service**; then click the **Start** button. Use the **Startup type** drop-down list to set the service to Manual, Automatic, or Disabled.

To use the Notify Service Manager:

- 1 Open the Notify Service Manager by choosing **Start | Programs | Merant | Merant Tracker | Notify Service Manager**. The Notify Service Manager appears in the form of a tree-view pane and a summary pane.



- 2 Expand the tree to see server nodes and project nodes. To add a server to the list, right-click Tracker Notify Service at the top of the tree; then choose **New server definition**.

Right-click a node to see a context menu, or select a server in the tree-view pane to see the projects listed in the summary pane. If a project is disabled, the word “disabled” appears in the Status column.

To export a tab-delimited text version of the Project list to a file, select a server; then choose Action | Export List. Alternatively, the Export List option in the context menus exports whatever is in the right pane—a list of either servers or projects—to a file.

NOTE If you create a new project through Tracker Administrator, Notify is disabled for that project by default. If you want to enable Notify for the new project, use the Notify Service Manager.

Stopping and Restarting Notify

From the Windows Service Manager: Click Stop, wait a moment for the service to stop, and then click Start.

From the Notify Service Manager: Right-click Tracker Notify Service in the tree view and then choose Stop or Start from the context menu. (You can also use the Start and Stop icons near the top of the window: the triangle for start, the square for stop.)

NOTE You must restart Notify Service for configuration changes to take effect.

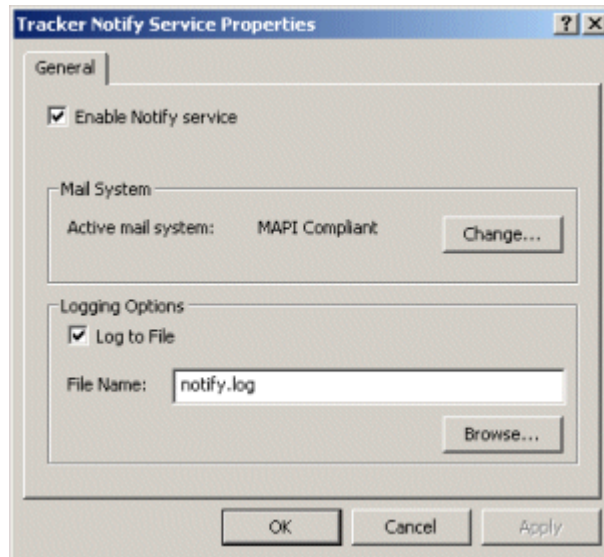
Changing Your Log-On Parameters

When you installed Tracker, either you provided a username and password for Notify Service to use to log in the system, or you left those fields blank indicating that the service should run under the Local System account. (If you use MAPI or Microsoft Mail, Notify Service must run under a user account.)

You can change this log-on information from the Log On tab of the Properties dialog box for Tracker Notify Service (double-click the service in the Windows Service Manager). If Notify Service runs under a user account, you must update the password here whenever it changes.

Configuring Global Properties

At the top of the tree-view pane in the Notify Service Manager is the Tracker Notify Service node. Choosing **Properties** from this node's context menu opens the Tracker Notify Service Properties dialog box.



On the **General tab**, you can enable or disable Notify Service.

Under **Mail System**, click Change to configure your e-mail system for use with Notify.

NOTE Notify Service will not run if you do not have an e-mail server configured.

Under **Logging Options**, you can enable or disable logging and choose the log file directory.

NOTE Log files are never automatically deleted. It is up to the user to remove or otherwise manage the log files.

CAUTION! When you're done, be sure to click OK to save any changes you make to the Notify properties.

NOTE You must restart Notify Service for configuration changes to take effect.

Configuring Server Properties

To view and modify the properties for a specific server, right-click the server name in the tree-view pane of the Notify Service Manager, and choose Properties from the pop-up menu. This opens the Server Properties dialog box.

On the **General tab**, you can enable notifications. Click **Mail Message Options** to configure your mail system for use with Notify.

CAUTION! When you're done, be sure to click OK to save any changes you make to the server properties.

NOTE You must restart Notify Service for configuration changes to take effect.

Configuring Your Mail System

See also

[“Changing Your Log-On Parameters” on page 264](#)

You need to configure Notify to provide the information needed to communicate with your mail system before you can send e-mail to other users on your project.

To configure your mail system:

- 1 Right-click Tracker Notify Service in the tree view, and choose Properties; then click Change under Mail System. The Mail System dialog box appears.
- 2 Select your mail system from the list. All mail systems that are supported by Notify appear in the list. If you choose **Auto Select**, Notify will select your mail system (If you have multiple mail systems set up, Notify will select the first one it finds).

Your active mail system appears by name below the drop-down list in the Mail System dialog box.

- 3 Choose **Configure** to set up your mail system.
- 4 After you have configured your mail system, click OK for your settings to take effect.

NOTE You must restart Notify Service for configuration changes to take effect.

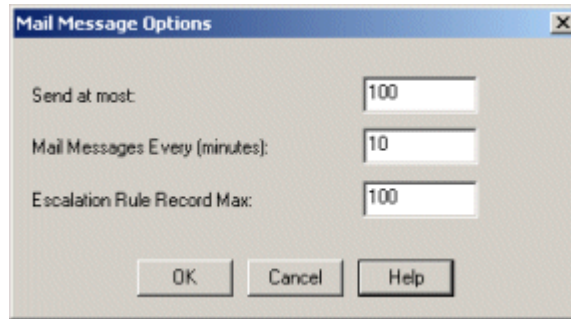
Setting Up Mail Message Options

When you start Notify, it scans all projects you have selected for notifications that need to be delivered. Notify transfers these notifications from the project database to your e-mail system. The number of messages sent at one time and the interval for re-scanning the database are controlled by the Mail Message Options settings.

You can change mail message options for each defined server in the Mail Message Options dialog box. The new options take effect the next time you start Notify.

To change mail message options:

- 1 Right-click the server in the Notify Service Manager, choose Properties from the pop-up menu, click Mail Message Options, and enter your server administrator password. The Mail Message Options dialog box appears.



- 2 Set the maximum number of messages to be sent at each interval (from 1 to 30,000) and the interval (in minutes) for sending messages (from 5 to 30,000).

You can set the number of notifications to a low value to control the amount of traffic that is put on your network at one time. Set the interval for sending messages to a high value if the projects selected have infrequent updates or if you are running Notify on a user's machine.

The interval for Notify's next scan for updates or changes starts after the previous scan is complete.

- 3 Enter the maximum number of records to be returned by an Escalation Rule query. The default is 100. If an Escalation Rule query returns more than the specified number of records, the query terminates, and an error is recorded in the log.

NOTE Large numbers of records can result in slow performance.

- 4 Click OK.

NOTE You must restart Notify Service for configuration changes to take effect.

14 Tracker Project Management and Recovery

In this Chapter

For information about . . .	See page . . .
Overview	272
Performing Data Backups and Recovery	272
Re-Indexing Database Tables	274
Preserving Database Integrity	275
Moving an Existing Database	283
Rolling Over an Existing Project Database	284
Controlling Record Locking	284
Controlling Logins	286
Improving Query Performance	290

Overview

Who Should Read This Chapter?	This chapter is intended for Server Administrators. Project Administrators may be interested in “Verify and Repair (Level 2 Corruption)” on page 278 .
Why Read This Chapter?	<p>This chapter describes standard database management functions as they relate to Tracker. These include:</p> <ul style="list-style-type: none">■ Backup and recovery■ Re-indexing database tables■ Preserving and repairing database integrity■ Moving and rolling over databases■ Managing locks■ Managing logins and login sessions

Performing Data Backups and Recovery

Each time a user submits an issue, saves a query or report, or changes the project configuration, Tracker must update several tables in the database. If a system failure occurs during this update, the database will be compromised and data lost. These crashes might occur as a result of a power failure, hardware failure, or a system software failure that requires rebooting the server or the workstation.

Backup Recommendations

Though recovery management routines can help restore your database, they cannot recover from the effects of system failures that destroy part or all of the database. Therefore, to ensure that

your data is safe, we recommend that you institute a regular project backup schedule. A good strategy is to make a full backup of the \TRACKER directory every week, with incremental backups daily. You may also want to save one full backup every month as an archive.

DOS Commands

Using the DOS commands on Tracker files carries some risk. If you mistype a command or the system crashes, you could lose data. Therefore, in addition to your regular backups, you should back up *TRKMASTER* before making major changes (like moving Tracker or creating a project template). You should perform scheduled backups when all users are logged off the system. See [“Controlling Logins” on page 286](#).

Backing Up Tracker Data

You will need to use your DBMS tools to do Tracker data backups. If your DBMS tools require all users to be out of the database during a backup, follow the instructions in [“Controlling Logins” on page 286](#).

Recovering Data from Backups

Only the Server Administrator can perform this procedure. Data will be recovered from the most recent backup.

Use your database management tools to overwrite the current *TRKMASTER* and all affected databases or userspaces with the backup. If your DBMS supports logging, restore data changes since the last backup from the log file.

Re-Indexing Database Tables

Tracker Administrator can re-index your database tables to maintain the speed of the lookups on indexed fields. Use this tool whenever Tracker performance starts to degrade. This tool:

- Deletes the existing *TRKMASTER* indexes
- Rebuilds the indexes

NOTE When you re-index tables, all users must be logged off the server. Disable new logins, force current users off the server, and remove any lost connections from the server *before* you re-index.

To...	See page...
Disable logins	287
Force current users off the server	288
Remove lost connections	289

To re-index the database:

- 1 Log in as the Server Administrator. Project Administrators cannot perform this procedure.
- 2 Select Manage | Deny Logins to Server to lock all users out of the server.
- 3 Select Tools | Who to display the user IDs of everyone on the system.
- 4 Select Manage | Force Off All. Users are reminded every minute for five minutes to log out of the system.
- 5 Once all users are out of the system, choose Tools | Re-Index Database.
- 6 Select the projects you want to re-index. To select more than one, hold down the SHIFT key while selecting each project that you want to include.

- 7 Select Re-Index Tracker Master.
- 8 Click OK.
- 9 Select Manage | Allow Logins to Server to allow users access to the server again.

Preserving Database Integrity

Database errors, abnormally terminated sessions, and the inability to log in to a Tracker project, may indicate database corruption. Using the Verify and Repair tool in Tracker Administrator ensures the integrity of your Tracker databases by reconstructing corrupted tables and replacing lost ones. Use the Verify tool regularly to ensure the database is in a consistent state. Use the Repair tool *only* if the Verify tool indicates that problems exist. *Always* make a backup of your database before repairing; then save the repair log file.

There are two levels of Verify and Repair:

- Critical verify and repair (Level 1 Corruption)
- Verify and Repair (Level 2 Corruption)

NOTE Verify and Repair are not backup tools. If you lost data, Verify and Repair will not replace it. You must restore lost data from your system backups. See [“Recovering Data from Backups” on page 273](#).

Before You Begin

When you run verify or repair, all users must be logged off the server. Be sure to disable new logins, force current users off the

server, and remove any lost connections from the server *before* you run verify or repair.

To...	See page...
Disable logins	287
Force current users off the server	288
Remove lost connections	289

Critical Verify and Repair (Level 1 Corruption)

Only the Server Administrator can perform this procedure.

Use the Critical Verify and Repair tools if you or your users are unable to log in to Tracker or Tracker Administrator. A Critical Verify and Repair looks at the basic tables and data essential for valid Tracker and project databases. Only the Server Administrator may do a Critical Verify and Repair, using the server administrator password.

What Critical Verify and Repair Checks

Critical Verify and Repair checks to make sure that the:

- *TRKMASTER* database exists
- *TRKMASTER* database contains these tables:

trkctl	trkprj	trksty
trkfldAlias	trkproc	trkxrel
trkglb	trkreg	
trkisType	trkrelalias	

- *trkglb* table contains the proper information

- Project database contains these tables:

trkassoc	trkisType	trkrul
trkch	trkmce	trkscrsl
trkchg	trkmod	trkscrst
trkctl	trknote	trksty
trkdep	trkperm	trkterm
trkfe	trkqry	trktfe
trkfile	trkreg	trktse
trkfld	trkrel	trktype
trkfme	trkrelIsType	trkug
trkfol	trkrelType	trkuge
trkform	trkrset	trkusr
trkglb	trkrpt	trkxprj

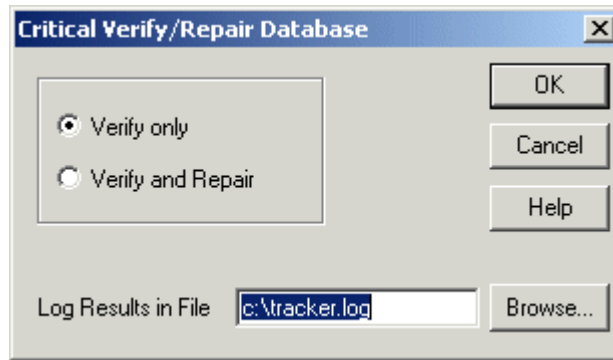
- Necessary columns for the database exist as described in the *Tracker Database Schema Specification* in the Tracker Administrator online help.

How to Use Critical Verify and Repair

To do a Critical Verify and/or Repair:

- 1 Open Tracker Administrator but do not log in. Click the Cancel button in the Open Project dialog box.
- 2 Select Tools | Verify/Repair Database. The Merant Tracker Administrator Login dialog box appears.

- 3 Login as Server Administrator. The Critical Verify/Repair dialog box appears.



- 4 Enter the path for the log file into which Verify and Repair will place a record of the verify/repair session. If errors are found, they will be displayed in this log file.

NOTE The first time you use Verify and Repair, the default path for the log file is C:\TRACKER.LOG. You can change the path, and Verify and Repair will remember the path you entered the next time.

- 5 Select Verify only to check for errors. Select Verify and Repair to repair the database if errors are indicated.
- 6 Click OK.

Verify and Repair (Level 2 Corruption)

The Server Administrator can verify and repair any project. Project Administrators can verify their own projects, but cannot repair them. However, Project Administrators can be given the ability to repair their projects' databases. (See "Permitting Repairs by Project Administrators" on page 281.) In all cases, the

Project Administrator can verify or repair only those projects they belong to.

Use the Verify and Repair tools when users report database errors and abnormally terminated sessions. A Verify and Repair at this level can look at potential sources of database corruption of both project and Tracker Master databases. To use Verify and Repair (Level 2 Corruption), you must be able to login to Tracker Administrator. If you cannot, do a Critical Verify and Repair.

What Verify and Repair Checks

Verify and Repair checks tables for the areas you select in the Verify/Repair Database dialog box. (See the procedure on page 280.)

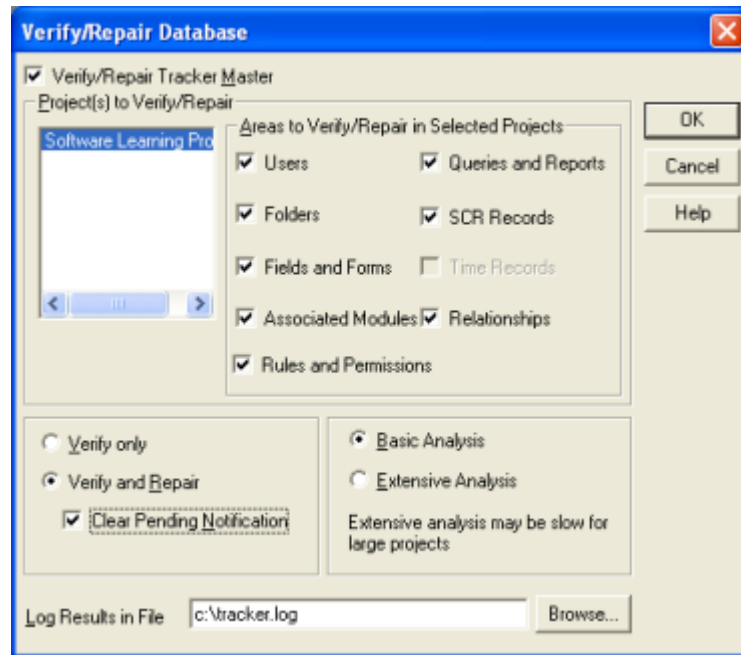
When you select the box for...	Verify/Repair checks these tables...
Verify/Repair Tracker Master	trkctl, trkprj
Users	trkuser, trkug, trkuge
Folders	trkfol, trkfe
Fields and Forms	trkfld, trkch, trkform, trkfme, trkterm
Associated Modules	trkassoc
Rules and Permissions	trkrul, trkperm
Queries and Reports	trkqry, trkrpt
Issue Records	trkscrsl, trkscrst, trkchg

For all selected projects, Verify/Repair checks the trkctl, trktype tables. Errors in trktype are corrected even in verify only mode, because this table is needed to run Verify/Repair.

How to Use Verify and Repair

To Verify and/or Repair:

- 1 Log in to Tracker Administrator as Server or Project Administrator.
- 2 Select Tools | Verify/Repair Database. The Verify/Repair Database dialog box appears.



- 3 Select the project you want to check from the Select Project(s) to Verify/Repair group.
- 4 Select which items to verify/repair from the Areas to Verify/Repair in Selected Projects group.
- 5 Select Verify only or Verify and Repair. If you choose Verify only and errors are indicated, do this procedure again choosing Verify and Repair to repair the problems.

NOTE If the *trkctl* table for the project database is corrupted, Verify will produce an error message in your TRACKER.LOG file about possible excessive user notifications. Repairing may cause notifications to flood users' mailboxes and In Trays. To avoid this, you can select the "Clear Pending Notifications" box when you choose Verify and Repair. All notifications will begin again from scratch.

- 6 Select Basic Analysis for routine verify/repair or Extensive Analysis for a more thorough verify/repair. On large projects, an Extensive Analysis will take more time than a Basic Analysis.
- 7 Enter the path for the log file into which Verify and Repair will place a record of the verify/repair session. If errors are found, they will be displayed in this log file.

NOTE The first time you use Verify and Repair, the default path for the log file is C:\TRACKER.LOG. You can change the path, and Verify and Repair will remember the path you entered the next time.

- 8 Click OK.

Error Messages

Verify and Repair error messages are explained in Troubleshooting Tips in the Tracker Administrator online help.

Permitting Repairs by Project Administrators

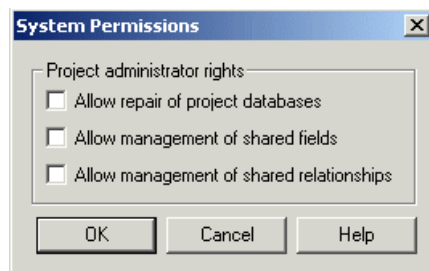
Only the Server Administrator can perform this procedure.

By default, the Server Administrator can perform Verify and Repair and Critical Verify and Repair. Project Administrators can only verify their own project. The Server Administrator can allow Project Administrators to repair their projects if desired, by using the following procedure.

How to Change Permissions

To enable or disable Project Administrator repair permissions:

- 1 Log in to Tracker Administrator as Server Administrator.
- 2 Select Tools | System Permissions. The System Permissions dialog box appears.



- 3 Check the Allow project administrator to repair project database box.
- 4 Check Allow management of shared fields and/or Allow management of shared relationships to expand project administrator permissions to be able to work with shared issue types and relationships across project boundaries.
- 5 Click OK.

Permissions will be set globally, for all Project Administrators. It is not possible to set this permission for specific users or projects.

Moving an Existing Database

To copy a database of one type to a database of another, use the procedure, [“Creating a New Project Based on an Existing One”](#) on page 100.

Rolling Over an Existing Project Database

Use this procedure to start a new project containing only the open issues of an existing project. This can help free up disk space if you have a lot of closed issues in your project. This procedure creates a new project based on the existing project.

To roll over a database:

- 1 Click the Projects button and open the project you want to roll over.
- 2 Follow the steps in [“Creating a New Project Based on an Existing One” on page 100](#).
- 3 Start Tracker.
- 4 Open the project you cloned in [step 2](#).
- 5 Select Query | Run and select All Closed.
- 6 Select Update | Delete, and select all the records in the dialog box.
- 7 Click OK.

Controlling Record Locking

A *record lock* prevents two users from updating the same record simultaneously. Tracker automatically places a lock on a record when a user begins updating it and clears it when they finish.

In certain situations, you may need to clear record locks manually. For example, if a user begins an update but does not complete it, the lock remains on the record and prevents other users from updating it. Using this procedure helps you determine who currently has locks on records and shows you how to clear them.

To clear record locks:

- 1 Click the Who button.
- 2 To clear record locks, choose one of the following:

To clear...**Then do this...**

A single record lock for a user

Select a user in the **Current Processes** list and click the Clear Record Lock button.

All record locks showing in the **Current Processes** list.

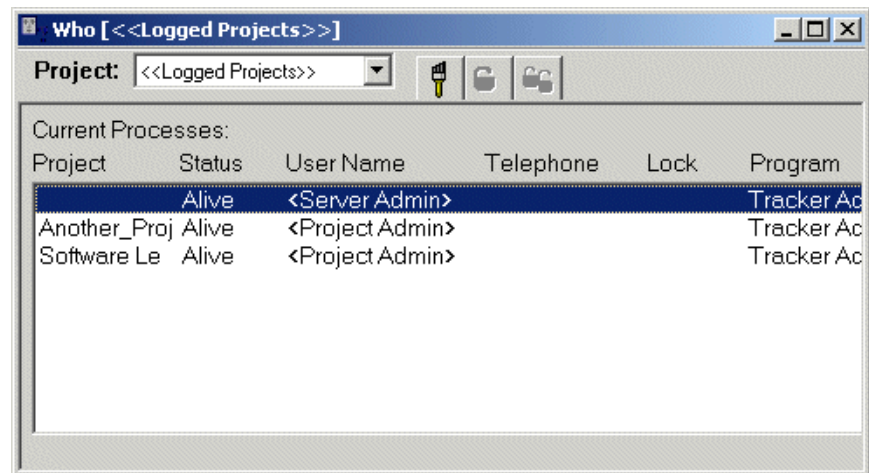
Choose Clear All Record Locks.

Turning Record Locking On or Off

The Server Administrator can turn record locking on and off.

To turn record locking on or off:

- 1 Click the Who button.



- 2 Select a project from the **Current Project** list in the toolbar.

- 3 Select one of the following:
 - Manage | Enable Record Locking on Update
 - Manage | Disable Record Locking on Update



Sample Scenario

The next morning, Brad's phone message light was flashing when he came in. One of the developers had just called to say that he was getting an error when he tried to access the Western project and modify SCR5. He knew the issue existed because he updated it yesterday, and he knew it wasn't in use because he was the only one logged in his department.

Brad recognized this as a record lock problem. He started Tracker Administrator and followed "[Controlling Record Locking](#)" on [page 284](#). The issue was not listed as being in use, so he released the lock and notified the developer who had called in. (This is the conclusion of the scenarios in this guide.)

Controlling Logins

Tracker Administrator enables you to control who is logged in to a project or a project server. You can deny access whenever you need to perform administrative tasks that require all users to be logged out, such as adding and removing fields, or backing up the database. You can also force present users off the project. These features are useful when you need to re-index or verify/repair a database.

Turning Off Logins

To deny logins to a server or project:



- 1 Click the Projects button.
- 2 Select the project you want to deny logins to.
- 3 Click the Who button.
- 4 If Who indicates there are users logged in to this project, select Manage | Force Off All. Wait for five minutes to allow Force Off to take effect.

NOTE Once you deny access to a server or project, Tracker Administrator displays a message to all users warning them to log out of the project or server. It repeats this message once every minute for five minutes, until it forcibly logs out any users who are still logged in.

- 5 To deny logins, choose one of the following:

To deny logins to a...	Select...
Project as a Project Administrator	A project from the Current Projects list. Then select Manage Deny Logins to Selected Project.
Project as the Server Administrator	A project from the Current Projects list. Then select Manage Deny Logins to Selected Project.
Server	Manage Deny Logins to Server.

Turning On Logins

To allow users to log back in to a server or project:

- 1 Click the Projects button.
- 2 To allow logins, choose one of the following:

To allow logins to a...	Then select...
Project	A project from the Current Projects list. Then select Manage Allow Logins to Selected Project.
Server	Manage Allow Logins.

Forcing Users Off Tracker

Occasionally it may be necessary to force users off the server or project so you can do database maintenance. Use the following procedure to force users off the system either with a five-minute warning, or in some emergencies, with no warning at all. Forcing off users might be necessary in the face of an impending power-down or if the DBA needs to bring the server down immediately.

To force users off the system:



- 1 Click the Who button.
- 2 In the Projects list, select the project (or choose All Projects) you want to force users off of.

- 3 If users are logged in to the project(s) you have selected, choose one of the following:

To force off...**Then do this...**

Selected users

- a Select the users you want to force off.
- b Select Manage | Force Off Selected.
- c Wait five minutes to allow Force Off to take effect.

All users

- a Select Manage | Force Off All.
- b Wait five minutes to allow Force Off to take effect.

Removing User Processes

Occasionally it may be necessary to remove user processes, for example, when a connection is lost, when there is an impending power down, or if the DBA needs to bring the server down immediately. Use the following procedure to remove user processes.

NOTE If you aren't sure whether the session is dead, Choose Who and look at its status. A status of Sleeping: 2 or greater indicates the session is dead or not running properly. For more information about statuses in the Who window, select Help | Search, and enter the keywords *who window*.

To remove user processes:



- 1 Click the Who button.
- 2 In the Projects list, select the project (or All Projects) for which you want to remove user processes.

3 If Who indicates there are users logged in to the project(s) you have selected, choose one of the following:

To remove...

Then do this...

Selected users
immediately

- a Select the users you want to force off.
- b Select Manage | Kill Now.
- c Click Yes.

All users immediately

- a Select all users by clicking them all.
- b Select Manage | Kill Now.
- c Click Yes.

Users according to their
Sleep status

- a Select Tools | Remove Lost Connections.
- b Select the **Enable Removing Lost Connections** option.
- c Specify a number of Sleep minutes at which to remove the process.

NOTE Removing Lost Connections does not work if Tracker Administrator is not running.

Improving Query Performance

To improve the performance of Query list handling from within Tracker, delete unused public and private queries, and keep the

number of user groups to a minimum. See [“To remove a report’s permissions, select the report in the Report list and click Remove.” on page 148](#) for more information.

Appendix A: Disk Space Allocation

We recommend the following space allocations for Tracker projects. This does not include space you allocated during the installation for the Tracker application.

MS SQL Server, MSDE, and Sybase

For ...	Allocate ...
Data Device (for project data)	10 MB for the Tracker Master database, plus 20 MB per 1000 defects.
Log Device (for log data)	30% of data device space—25 MB if possible.

Oracle

For ...	Allocate ...
Default Tablespace	Unlimited.
Temporary Tablespace	Unlimited. This release of Tracker does not use Temporary Tablespace.

NOTE For more information, choose Help | Search and enter keywords disk space.

Appendix B: Default Directory Structure

This appendix describes the directory structure in a fresh installation of Tracker where all the defaults have been accepted during the installation.

This directory...	Contains...
\MERANT\TRACKER	Default Tracker installation. All the directories are under this directory.
\NT	Tracker and Tracker Administrator executables.
\MERANT\TRACKER\NT	Notify executable (if purchased).
\SYS	Network-wide configuration files
\TKNT	The setup application, which includes the SETUP.EXE and INSTALL.EXE. SETUP.EXE must be run on the user's workstation to complete a workstation installation and to add or reconfigure the ODBC drivers.
\TTK	Tracker Toolkit (TTK)
\DB	Learning Projects

For more information, see ["How Tracker Fits Into Your DBMS Environment" on page 38.](#)

Appendix C: Evaluating Tracker Using MSDE

Use the MSDE with Tracker for evaluation or training purposes only

Before you begin performing any tasks with the MSDE database available for use with Tracker, note that this DBMS is provided with the product *for evaluation or training purposes only*. This system is intended to be used only as a pilot testing tool for users evaluating Tracker or learning how to use Tracker with the Learning Projects.

For production data, use the other DBMSs supported by Tracker: MS SQL Server, Sybase, and Oracle.

NOTE If you installed the MSDE along with Tracker, the system administrator password for the MSDE is `PVCSTracker#1`.

You may begin testing Tracker immediately using the Learning Projects and MSDE (Microsoft SQL Server 2000 Desktop Engine). When you are ready to use Tracker on production data, you can convert any existing work you performed with MSDE for use with another DBMS by following the instructions in [“Creating a New Project Based on an Existing One” on page 100](#).

NOTE On your first login, Tracker automatically loads the Learning Projects, ready to open, for you to practice on. These projects contain sample data and run under MSDE.

Appendix D: Configuring the Web Client Interface

This appendix describes two files included in Tracker for using the web client interface to Tracker projects. The procedures described here are optional configuration steps.

The SAMPLE.HTM File

This file is provided to help you place elements of the Tracker web client functionality where you need them. During the installation, it is placed in the \Merant\Tracker directory. An icon for this file is placed in the program group called Tracker Web Client Sample Page. Code samples are provided for three features:

- Invoke the project configuration utility
- Log in to the Tracker web client as an authenticated user
- Anonymously submit a change request on the default project

In each case, HTML is provided to generate a button that will allow you to reach the particular feature described.

The WTSETUP.EXE File

This utility, located in the \Tracker\NT directory, is provided to assist you in reconfiguring the URLs and paths specified in your registry during the installation process. An icon for this application is placed in the program group and is called Tracker Web Client Configuration.

Index

A

- access
 - about 286
 - anonymous users 20
- adding
 - the project 89
 - users 20, 107, 110
- Admin login 68
- Admin user 28
- administrator
 - login 63
 - roles 39, 52
 - tasks 41
 - tasks (using password) 79
 - Tracker project 40
 - Tracker server 40
 - Tracker web client 40
 - types 41
- alias to view fields 203
- archives location 251
- archiving 273
- avoiding stumbling blocks 50

B

- backup
 - archiving 273
 - data backups 272
 - data recovery 272, 273
 - reattaching a database to Tracker 98
 - recommendations 272
 - with DBMS tools 273
- browser interface to Tracker 19, 22

C

- change description template
 - creating 253
 - customizing 254
- Change History view 182, 183
- Change Request View 181
- change text
 - augmenting 253
 - templates 253
- changing view options 26
- checklists
 - avoiding stumbling blocks 50
 - defining organizational goals in implementing Tracker 49

- defining roles 52
 - gathering system information 53
 - outlining purpose of Tracker 48
 - preparing the server 68
 - child field 229
 - choice fields
 - defining 215
 - restrict 228
 - choices, restrict 219
 - clearing record locks 285
 - complex notification rules 261
 - Conditional Notify 28, 260
 - configuration file
 - adding 255
 - codes 251
 - definition 251
 - locating archives 251
 - removing 255
 - configuring
 - mail options 267
 - Notify properties 264
 - server properties 266
 - constraints, for cross-project relationship types 204
 - contacting technical support 15
 - controlling record locking 284
 - conventions, typographical 14
 - copying
 - fields 234
 - users 119
 - corruption of database 275
 - creating
 - devices 65
 - projects 20, 85
 - projects with password 89
 - tables
 - Oracle 174
 - SQL 161
 - Sybase 168
 - views
 - Oracle 174
 - SQL 161
 - Sybase 168
 - critical verify and repair 276
 - cross-project relationship types
 - constraints 204
 - defined 201
 - cross-project tables
 - description 180
 - Oracle 175
 - Sybase 169
 - cross-project views
 - description 180
 - Oracle 175
 - Sybase 169
 - custom field
 - adding 214
 - creating 211
 - definition 28
 - custom forms 20
 - custom templates
 - creating 254
 - using 253
 - custom user groups
 - creating 149
 - deleting 151
 - customizing
 - fields 209, 211
 - forms 209
 - terminology 245, 246
 - terms 209
- ## D
- data
 - archiving 273
 - backups 272, 273
 - data device 91
 - recovery 272
 - data device 28, 91
 - data entry, simplify 228
 - data integrity, improve 228
 - data types available 212
 - database 64, 275
 - administrator 39
 - corruptions 278

- creating (Oracle) 66, 79
- critical verify and repair 276
- device 28
- ensuring integrity 275, 276, 278
- errors 275, 278
- management functions 272
- master 63, 81
- master for SQL Server and Sybase 65
- reattaching an existing database 98
- re-indexing 274
- repairing corruptions 275
- report writing (SQL) 158
- rolling over 284
- structure 88
- supported types and versions 38
- verifying 278
- database device 28
- database permissions
 - required for Report Assistant (Oracle) 173
 - required for Report Assistant (SQL) 160
 - required for Report Assistant (Sybase) 166
- database views 156
- datasources
 - about 64
 - defined 28
- date fields, defining 220
- DBA
 - definition 28
 - tasks 40, 79
- DBMS
 - adding users 107
 - definition of 28
 - login relationship with Tracker 66
 - storing Tracker data 38
- default login method 66
- defining
 - choice fields 215
 - date fields 220
 - number fields 223
 - organizational goals in implementing Tracker 49
 - relationship types 200
 - roles 52
 - string fields 225
 - time fields 221
 - user fields 226
- delete relationship types 205
- deleting
 - fields 232
 - projects 103
 - users 117
- deleting issue types 195
- dependent fields
 - configure 228
 - defined 28
 - token groups 150
- detail-node role 198
- development engineers 133
- device
 - about 64, 88
 - creating 65, 79
 - creating in Oracle 66
 - creating in SQL Server 7.0 65
 - creating in Sybase 65
 - data 28, 79, 80, 81, 91
 - log 29, 64, 91
 - space allocation 80, 81
 - transaction log 79, 80, 81
- dialog box
 - Mail Options 268
 - Notify Properties 264
- directives 251
- directories, created by Tracker 295
- disk init command (Sybase) 65
- disk space allocation
 - server 63
 - Tracker projects 293
- displaying
 - status bar 26
 - toolbar 26
- documentation
 - Merant Tracker Database Schema Specification 23
 - ordering hard-copy manuals 15

E

- e-mail
 - address 106
 - options 114
- e-mail messages
 - for which project(s) 268
 - how often sent 268
 - maximum number sent 268
 - where sent 268
- exchange data
 - between Merant Tracker and Merant Version Manager 250
 - between Merant Tracker and source control 250
- exiting
 - Report Assistant 165, 172, 178
 - Tracker Administrator 59

F

- features
 - Tracker 18
 - Tracker web client 18
- field permissions 135, 143
 - modifying 144
 - removing 144
 - setting 143
- fields 33
 - child 228
 - copying 234
 - customizing 209, 211
 - data types available 212
 - defining choice fields 215
 - defining date fields 220
 - defining number fields 223
 - defining string fields 225
 - defining time fields 221
 - defining user fields 226
 - deleting 232
 - dependent 228

- indexed 274
- limitations 213
- modifying 231
- parent 228
- recovering 233
- selecting for forms 236
- time-related calculations 183
- valid 186

- Final State 197
- forcing users off Tracker 288
- forms 236
 - customizing 209
 - editing 236
 - modifying 240
 - selecting fields for 236

G

- gathering system information 53
- general permissions 135

H

- hard-copy manuals, ordering 15

I

- illegal character treatment
 - Report Assistant 158
- implementing Tracker
 - checklists 47, 48, 49, 50, 52, 53
 - combining administrator roles 45
 - communication issues 45
 - corporate goals 45
 - effect on defect tracking environment 45
 - organizational size and structure 44
 - planning 44

- technological constraints 45
- indexed fields, lookups 274
- interface
 - main window 25
 - manage menu 26
 - task windows 27
 - toolbar 26
- introduction to Tracker Administrator 17, 18
- issue relationships 185, 197
- issue types
 - about 186
 - adding 188
 - creating 185
 - defining 187
 - deleting 195
 - lifecycle 186
 - about 192
 - defining 192
 - modifying 190
 - permissions 191
 - recovering 195
 - relationships 197
 - roles 197
 - sharing 193
 - system 193

K

- kill user processes 289

L

- Learning Projects, directory structure 295
- lifecycle 186
- list of projects 64, 87
 - adding a database 98
- log device 29, 91
- logging in
 - definition 30
 - Notify 262

- switching between projects 59
 - Tracker Administrator 56
- logging out 59
- login
 - abnormal terminations 275
 - Admin 68
 - allowing 286, 288
 - default method 66
 - denying 286, 287
 - for administrators 67
 - multiple login method 67
 - one-to-many relationship 66
 - one-to-one relationship 67
 - problems 275, 276
 - sa or system password 68
 - Tracker-DBMS relationship 63
 - trkadmin 68
 - turning on/off 287, 288
 - users 63, 66

M

- Mail Options dialog box 268
- main window 25
- manage menu 26
- managers, definition 133
- managing
 - locks 272
 - projects 85
- MAPI 30
- master database 63, 81
 - description 64
 - SQL Server and Sybase 65
- MASTER.DAT in SQL Server and Sybase 65
- master-node role 198
- Merant Tracker information flow 259
- Merant, contacting 15
- message options, Mail Options dialog box 268
- Microsoft SQL Server 38
- mode, definition 31
- modifying

- fields 231
- forms 240
- permissions 135
- query permissions 146
- report permissions 148
- users 116

MSDE

- supported version 38

- multiple login method 67

N

- network configuration files 295

- new project 85

- note titles

- adding 243
 - deleting 244
 - modifying 243

- notification

- concepts 258
 - process 260

- notification options

- description 260

- Notify

- definition 23, 31
 - functions 23
 - logging 265
 - using e-mail addresses 106, 112
 - where stored 295

- Notify Properties dialog box 264

- number fields 223

O

- ODBC

- definition 31
 - error 104
 - learning about 157

- one-to-many login relationship 66

- one-to-one login relationship 67

- online help

- accessing 14
 - for the command-line interface 14
 - for the GUI 14

- online manuals, ordering hard-copy manuals 15

- Oracle

- adding users 110
 - database creation 66
 - DBA tasks 63
 - running Report Assistant 173
 - storing Tracker data 38
 - structure 88
 - supported version 38
 - system password 63
 - tablespace creation 66
 - Tracker project disk space allocation 293
 - user ID 106
 - using Report Assistant 173

- ordering hard-copy manuals 15

- outlining purpose of Tracker 48

P

- parent field 228, 229

- password

- sa password 63, 68, 71
 - system password 71
 - system password (Oracle) 63
 - Tracker password 63
 - trkadmin password 68

- PCMS Dimensions Interchange File Format 31

- PDIFF 31

- performance

- improving 274, 275, 276, 278

- permission

- Add New Relationship 205
 - Change Issue Type 187
 - Delete Existing Relationship 205

- permissions

- administrator 133
 - assigning 134, 137, 142

- available categories 135
- definition 32, 132
- managing 131
- modifying 132, 144
- removing 137, 142, 144
- setting field 143
- setting general 137
- setting query 145
- setting record 141
- setting report 147
- verify and repair 281
- preparing the server
 - about 61, 63, 79
 - before you begin 68
 - prerequisites 62
- printed manuals, ordering 15
- program access 286
- project
 - adding 89
 - copying 100
 - creating 85, 87
 - with password 89
 - data 91
 - database 87
 - defined 32
 - definition 87
 - deleting 103
 - disk space allocation 293
 - list of projects 87
 - managing 85
 - new based on existing (project rollover) 284
 - reattaching an existing database 98
 - renaming 102
- project administrator 32, 39, 116
- project and log data 88
- PVCSTRKUSER 32
- PVCSTRKUSR 109, 110

Q

- quality engineers 133

- query
 - creating custom 21
 - definition 32
- query permissions
 - about 135, 145
 - modifying 146
 - removing 146
- querying views, rows returned 165, 172, 178

R

- reattaching an existing database 98
- record locking
 - clearing 285
 - controlling 284
 - turning on/off 285
- record numbering
 - controlling 244
 - description 244
 - setting the next record ID 244
- record permissions
 - available 135
 - hierarchy 139
- recover relationship types 205
- recovering
 - deleted fields 233
 - deleted users 118
- recovering issue types 195
- recovery management 272
- re-indexing database tables 274
- relationships
 - about 197
 - allowed types 199
 - bi-directional navigation 199
 - cross-project relationships 199
 - cross-project types 201
 - defining 200
 - delete 205
 - dependent 198
 - nested relationships 199
 - permissions 205
 - predefined types 198

- recover 205
- sharing 202
- system 202
- viewing fields in 203
- viewing peers 199
- removing
 - lost connections 290
 - query permissions 146
 - report permissions 148
 - user processes 290
- renaming projects 102
- repair
 - database 275, 276, 278
 - permissions 281
- repair, See verify and repair
- Report Assistant
 - about 156
 - creating project lists 179
 - creating tables (Oracle) 174
 - creating tables (SQL) 161
 - creating tables (Sybase) 167
 - creating views (Oracle) 174
 - creating views (SQL) 161
 - creating views (Sybase) 167
 - human readable tables 156
 - illegal character treatment 158
 - purpose of 156
 - query/report tool requirements 159
 - required database permissions (SQL) 160
 - required database permissions (Sybase) 166
 - skills required (SQL) 157
 - Sybase limitations 166
 - updating views/tables 180
 - views vs. tables 156
- report permissions
 - about 135
 - modifying 148
 - removing 148
 - setting 147
- restrict choice fields 228
- RK directory 295
- rolling over a project 284

S

- sa password
 - about 68, 71
 - definition 33
 - Tracker functionality 63
- schema
 - documentation 23
 - Tracker Administrator 158
 - userspace (Oracle) 35
- SCR
 - definition 33
- security 286
- server
 - before you begin 63, 68
 - definition 33
 - disk space allocation 63
 - pointing Tracker to devices 63
 - preparing 61, 63
 - by DBA 71
 - by Tracker server administrator 71
 - DBA 62
 - prerequisites 62
 - server administrator 62
- server administrator
 - definition 33, 39
 - login 63
 - password 68
 - tasks 41
 - tasks (using password) 79
- session termination, repairing database 278
- Setting Up LDAP Authentication 121
- share data
 - between Merant Tracker and source control 250
 - between Mernt Tracker and Merant Version Manager 250
- share relationship type 201
- shared issue types 193
- sharing relationship types 202
- single sign-on 33, 120
- source control, benefits of integrating with Tracker 251

SQL Server

- 7.0, creating devices 65
- adding users 109
- DBA tasks 63
- device recommendations 64
- running Report Assistant 160
- sa password 63
- structure 88
- supported versions 38
- Tracker project disk space allocation 293
- user ID 106
- using Report Assistant 160

SQL, storing Tracker data 38

standard field 33

status bar 26

string fields 225

submitters 133

support engineers 133

switching

projects 59

servers 59

Sybase

- adding users 109
- creating devices 65
- DBA tasks 63
- device recommendations 64
- sa password 63
- storing Tracker data 38
- structure 88
- supported version 38
- Tracker project disk space allocation 293
- user ID 106
- using Report Assistant 166

system administrator 39

system issue type 187

system issue types 193

system password 68, 71

definition 34

Tracker functionality 63

system relationship types 202

T

table

Change History 182

re-indexing 274

updating 180

tablespaces

about 63, 64, 88

creating 79

creating in Oracle 66

temporary 92

task windows 27

templates

creating 253

customized 253

terms

and the Tracker web client 248

customizing 209, 245, 246

resetting to default 247

time fields 221

token groups 229

toolbar 26

Toolkit, where stored 295

tr2vcn.dll 252

Tracker

benefits of integrating with source control 251

browser interface 19, 22

compatible databases 38

components of 22

directory structure 295

functionality 63

functions 22

implementation checklists 47

list of projects 64

master database 63

planning and implementing 44

server tables 34

tables 64

using two databases 39

where stored 295

Tracker Administrator

components 22

- custom forms 20
 - exiting 59
 - field names 158
 - functions 22
 - interface 25
 - introduction 17, 18
 - Learning Project 58
 - logging in 56
 - logging out 59
 - Report Assistant utility 156
 - roles 40
 - schema 158
 - switching to another project 59
 - switching to another server 59
 - where stored 295
 - Tracker web client
 - administrator 40
 - anonymous user access 20
 - custom queries 21
 - features 18
 - functions 22
 - limitations 20
 - project configuration 19
 - Tracker-DBMS login relationship 66
 - TrackerLink functions 23
 - transaction log data 91
 - transition field
 - defined 213
 - setting up 219
 - transition lifecycle 186
 - TRKMASTER 63, 64, 81
 - definition 34
 - list of projects 87
 - trkvcn.dll 252
 - turning record locking on/off 285
 - typographical conventions 14
 - updating views/tables 180
 - user fields 226
 - user groups
 - administrator 133
 - copying 152
 - definition 34, 107, 132
 - deleting custom 151
 - development engineers 133
 - managers 133
 - managing 131
 - modifying permissions 132, 135
 - public 133
 - public queries/reports 133
 - quality engineers 133
 - selecting a group to modify 136
 - setting permissions 132–148
 - submitters 133
 - support engineers 133
 - user IDs
 - Admin 81
 - administrators 63
 - associated information 106
 - one-to-many 66
 - one-to-one 67
 - Oracle 106
 - SQL Server 106
 - Sybase 106
 - Tracker 81
 - Tracker and DBMS 66
 - users 63
 - users
 - about 106
 - adding 107, 110
 - allowing login 288
 - clearing record locks 285
 - controlling record locking 284
 - copying 119
 - deleting 116, 117
 - denying login 287
 - forcing off 288
 - login 63
 - modifying 116
 - recovering deleted 118
 - turning record locking on/off 285
- ## U
- UNION of views
 - SQL 162
 - Sybase 169

- userspace 35
- utilities, Report Assistant 156

V

- valid field 186
- Vender Independent Messaging 35
- verify and repair
 - definition 35
 - Level 1, when to use 276
 - Level 1, who can use 276
 - Level 2, when to use 279
 - Level 2, who can use 278
 - project administrator permissions 281
- verifying and repairing database 275, 278
 - critical 276
- version label 253
- Version Manager
 - benefits of integrating with Tracker 251
- view options 26
- VIM 35
- virtual tables 156

W

- web client 18
- windows
 - arranging 27
 - main window 25

